MICROSOFT.

MS™-DOS VERSION 3

Printed in Germany



SOFTWARE-LIZENZ-VEREINBARUNG
für Endbenutzer

WICHTIG: Diese Programme sind Eigentum der beiden Firmen Microsoft Corporation (nachfolgend Microsoft genannt) und ZENITH DATA SYSTEMS CORPORATION (nachfolgend ZDS genannt). Sie werden durch das Urheberrechtgesetz geschützt. Diese Programme werden nur zum Gebrauch für **einen** Mikrocomputer in Lizenz vergeben. Sie werden nicht verkauft. Die Lizenz wird nur unter der Bedingung vergeben, daß Sie mit dieser Mitteilung für Endbenutzer und dem SOFTWARE-LIZENZ-ABKOMMEN einverstanden sind. Lesen Sie sich dieses Abkommen bitte sorgfältig durch. Sind Sie mit den Bedingungen nicht einverstanden, geben Sie den beigefügten, ungeöffneten Umschlag mit den Disketten an Ihren Händler zurück, ebenso wie die mitgelieferten Handbücher. Sind Sie mit den genannten Bedingungen einverstanden, unterschreiben Sie die ausgefüllte Rückseite und senden diese Vereinbarung sofort zurück.

JEGLICHER GEBRAUCH DER PROGRAMME OHNE VORHERIGE RÜCKSENDUNG DER VEREINBARUNG IST UNGESETZLICH.

Unter der Berücksichtigung der von Ihnen gezahlten Lizenzgebühr und gemäß der Lizenzvereinbarung zwischen ZDS und Microsoft, gewährt ZDS auch Ihnen eine Sublizenz. Sie dürfen damit die Disketten und die beigefügte Dokumentation (nachfolgend SOFTWARE genannt) gemäß den folgenden Bedingungen benutzen:

Sollten Sie mit den Verpflichtungen dieser Vereinbarung nicht einverstanden sein, ÖFFNEN SIE NICHT DEN VERSIEGELTEN UMSCHLAG MIT DEN DISKETTEN. Geben Sie den Umschlag und die beigefügten Handbücher Ihrem Händler zurück.

Sie dürfen die Programme nur benutzen, nachdem Sie die Vereinbarung unterschrieben ZDS zurückgesandt haben. Sie dürfen die SOFTWARE Dritten in keiner Weise zugänglich machen.

Sie verpflichten sich, die SOFTWARE vertrauensvoll zurückzusenden, außer allgemein veröffentlichten Bedieneranleitungen, die Sie von Microsoft oder ZDS erhalten haben, oder die allgemein veröffentlicht wurden, oder die Sie gesetzlich von Dritten erworben haben, oder die Sie schon vor Erhalt dieser SOFTWARE in Besitz hatten.

Sie haben nicht das Recht, in irgendeiner Art die SOFTWARE zu vervielfältigen, weder in Teilen noch als Ganzes. Mit einer Ausnahme: Sie dürfen maximal **zwei** Kopien der Disketten anfertigen. Diese Kopien dürfen nur zu Sicherheitszwecken und zur Benutzung für **einen** Mikrocomputer angefertigt werden. Sie dürfen die Programme nicht zu einem anderen Computer mittels Kabel übertragen.

Die mitgelieferten Handbücher und das begleitende Dokumentationsmaterial darf nicht kopiert oder auf andere Art vervielfältigt werden. Die SOFTWARE darf von einem Computer zu einem anderen System übertragen werden. Sie darf aber unter keinen Umständen auf beiden Systemen gleichzeitig benutzt werden.

Sie sind damit einverstanden, über die angefertigten Kopien einen Nachweis zu führen. Dieser Nachweis beinhaltet das Datum der Kopie, den Zweck der Kopie und den momentanen Aufbewahrungsort der Kopie. Sie sind ebenfalls damit einverstanden, alle angefertigten Kopien vollständig mit allen Urheberrechtsvermerken der Originale und sonstigen Notizen der Originaldisketten zu versehen.

Sobald Programmüberarbeitungen oder Erneuerungen verfügbar sind, verpflichtet sich ZDS, Ihnen dies mitzuteilen. Diese Mitteilung erfolgt nur an registrierte Lizenznehmer.

Dieses Abkommen tritt mit dem Tage der Unterzeichnung durch Sie in Kraft. Es bleibt solange gültig, bis es widerrufen wird. Die SOFTWARE darf nicht benutzt werden, solange Sie dieses Abkommen nicht unterschrieben zurückgesandt haben.

Sollte das Abkommen von ZDS oder Microsoft beendet werden, weil Sie dagegen verstoßen haben, verpflichten Sie sich, alle Originaldisketten und alle Kopien, auch Teilkopien an ZDS zurückzusenden.

Diese SOFTWARE kann technische Daten beinhalten, die unter die Exportbeschränkungen der US-Regierung fallen. Sie verpflichten sich daher, diese SOFTWARE nicht weiter zu exportieren, weder direkt, noch indirekt durch Dritte, auch nicht durch Vergabe weiterer Lizenzen. Sie verpflichten sich, die entsprechenden Exportbeschränkungen der US-Regierung einzuhalten.

ACHTUNG

DIESE SOFTWARE WIRD IN LIZENZ VERGEBEN UND NICHT VERKAUFT. FÜR DIESE LIZENZ DÜRFEN AN DEN ENDBENUTZER SUBLIZENZEN VERGEBEN WERDEN. DAMIT SIND KEINE GARANTIEN IRGENDWELCHER ART VERBUNDEN. MICROSOFT CORPORATION UND ZENITH DATA SYSTEMS VERGEBEN MIT DER LIZENZ KEINE GARANTIE, DASS DIESE PROGRAMME EINWANDFREI ARBEITEN. EBENSO IST MIT DER WEITERGABE DER PROGRAMME KEINE AUSSAGE DARÜBER VERBUNDEN, DASS KEINE PATENTE ODER RECHTE DRITTER VERLETZT WERDEN.

MICROSOFT.

MS™-DOS VERSION 3

Printed in Germany

ZENITH data systems

ACHTUNG
Diese **Programme** werden in Lizenz vergeben (nicht verkauft). Für diese Lizenz dürfen an den Endbenutzer Sublizenzen vergeben werden. Damit sind keine Garantien irgendwelcher Art verbunden.

Die Besitzer und Distributoren vergeben mit der Lizenz an den Endbenutzer keine Garantie, daß diese Programme einwandfrei arbeiten. Ebenso ist mit der Weitergabe der Programme keine Aussage darüber verbunden, daß keine Patente oder Rechte Dritter verletzt werden.

In diesem **Handbuch** können technische oder orthografische Fehler enthalten sein. Änderungen werden laufend herausgegeben; neuere Auflagen dieses Handbuches enthalten diese Änderungen bereits.

Bei **technischen Problemen** wenden Sie sich bitte zuerst an Ihren Händler. Wir helfen Ihnen ebenfalls gerne weiter, wenden Sie sich bitte an:

ZENITH DATA SYSTEMS GmbH
Robert-Bosch-Straße 32-38
Postfach 10 20 60
6072 Dreieich/Sprendlingen

Warenzeichen und Copyrights

Mindestausstattung für MS-DOS Version 3:

a. Zwei Disketten 5.25 Inch, DS/DD 48-tpi, soft-sektoriert
b. Zenith PC, 128Kb RAM, 1 Diskettenlaufwerk, Bildschirm

ZENITH DATA SYSTEMS GmbH
Dreieich-Sprendlingen
im Mai 1986

Übersetzt von: Michael Werf[illegible]
Satz und Druck: Aschendorff, Münster

Inhaltsverzeichnis

Einführung

Teil I: Vorbereitung

Kapitel 1 **Erste Begriffe**

Inhaltsverzeichnis

Teil II: Grundzüge MS-DOS

Inhaltsverzeichnis

Inhaltsverzeichnis

Teil III: Befehlserklärungen

Teil IV: Weitere Dienstprogramme

Inhaltsverzeichnis

Teil V: Dienstprogramme für Winchesterlaufwerke

Teil VI: Anhang

Einführung

Mit diesem Handbuch erhalten Sie die aktuelle Version des Microsoft-Betriebssystemes MS-DOS, die Version 3.10. Auf Grund der zahlreichen Dienstprogramme erhalten Sie zwei Disketten. MS-DOS ist das gebräuchlichste Betriebssystem für Ihren Zenith PC.

In diesem Handbuch werden alle Funktionen und Befehle der Version 3.10 ausführlich erläutert. Das beigefügte Einführungsheft erklärt dem neuen Computerbesitzer die wichtigsten Funktionen in leicht verständlicher Form.

MS-DOS Version 3.10

Auf der Systemdiskette eins sind die eigentlichen Systemprogramme sowie weitere Dienstprogramme enthalten. Auf der Systemdiskette zwei befinden sich besondere Dienstprogramme für Winchesterlaufwerke und zur Programmherstellung.

MS-DOS Version 3.10 ist eine erweiterte Fassung des bisherigen Betriebssystemes MS-DOS 2.11. Die Version 3.10 enthält mehr Befehle und Dienstprogramme, die das System flexibler und komfortabler gestalten. Zusätzlich enthält es bereits alle Funktionen für die Installation eines Netzwerkes. Die Kapitel in diesem Handbuch erklären alle Einzelheiten sehr ausführlich. Der folgende Abschnitt gibt eine Übersicht der neuen Befehle und Programme.

Einführung

Neue Befehle

ATTRIB	zeigt und ändert das Nur-Lesen-Attribut einer Datei. Dieses Attribut erzeugt einen Softwareschutz, der ein Beschreiben oder Löschen der Datei nicht gestattet.
FCBS n,m	Befehl der Datei CONFIG.SYS. Bestimmt mit **n** die Zahl der gleichzeitig durch einen FCB (File Control Block) geöffneten Dateien; bestimmt mit **m** die Zahl der zu schützenden FCBs vor einer zwangsweisen Schließung durch DOS.
GRAFTABL	bestimmt die Zeichen mit den ASCII-Werten 128–255 für den Farb- oder Grafikadapter.
JOIN	ersetzt eine Laufwerkbezeichnung vorübergehend durch einen Verzeichnisnamen. Damit wird das gesamte Laufwerk einschließlich aller Unterverzeichnisse wie ein Inhaltsverzeichnis behandelt.
KEYBxxxx	Programm zur Erzeugung einer landesspezifischen (deutschen) Tastatur.
LABEL	erzeugt, verändert oder löscht den Namen einer Diskette.
LASTDRIVE	Befehl der Datei CONFIG.SYS. Bestimmt das letzte verfügbare Laufwerk.
SELECT	wählt das Programm zur Tastaturanpassung (KEYBxxxx) und den landesspezifischen Datensatz (COUNTRY) aus.
SHARE	Programm zur Unterstützung von Netzwerk und File-Sharing. Benutzt die neuen Funktionen von MS-DOS 3.10. Zusätzlich benötigen Sie einen Netzwerkadapter, MS-NET und die entsprechenden Programme.
SUBSTITUTE	ersetzt ein Inhaltsverzeichnis vorübergehend durch einen Laufwerknamen. Damit können auch ältere Programme, die ursprünglich für DOS 1.1 oder 1.25 geschrieben wurden, das strukturierte Inhaltsverzeichnis benutzen.

Einführung

Erweiterte Programme

COMMAND/F	veranlaßt das Betriebssystem nach einem Plattenfehler direkt in das Anwenderprogramm zurückzugehen. Damit bleibt das Programm mit allen Daten erhalten.
DISKCOMP **DISKCOPY**	Die beiden Programme können jetzt auch die neuen Laufwerke mit einer Speicherkapazität von 1,2 Mb oder mehr bearbeiten.
FORMAT	Es sind zwei weitere Zusätze möglich. Der Zusatz /4 formatiert in den neuen Laufwerken Disketten mit einem Format von 320 oder 360 Kb. Der Zusatz /B reserviert Speicherplatz für die Systemdateien. Damit ist später eine problemlose Übertragung mit dem Befehl SYS möglich.
LIB	Der neue Zusatz /P bestimmt die Mindestgröße einer Seite. Zusätzlich wird bei jeder Änderung automatisch eine Sicherungskopie der alten LIB-Datei erstellt.
LINK	Es sind sieben neue Zusätze möglich. Damit können z. B. auch einzelne Module als Overlay's bestimmt werden. LIB-Dateien können in maximal 16 verschiedenen Verzeichnissen gesucht werden.
PRINT	Es sind sechs neue Zusätze erlaubt, die eine größere Flexibilität des Programmes ermöglichen.

Befehlseingabe

Bei der Befehlseingabe für externe Dienstprogramme kann außer einer Laufwerkbezeichnung jetzt auch ein Verzeichnisname vorangestellt werden. Aus dem Grunde ist ein Punkt als Kurzeingabe für den Befehl REM nicht mehr gestattet.

Einführung

Allgemeine Änderungen

Um mehr Kompatibilität zum PC-DOS zu erreichen, wurde das Programm MAP in ASSIGN umbenannt. Damit arbeiten verschiedene Anwenderprogramme, die auf das Programm ASSIGN zurückgreifen, jetzt problemlos. Das bisherige Programm ASSIGN wurde in ASGNPART umbenannt.

Ebenso wurden verschiedene Zusatzeingaben für die Programme BACKUP und RESTORE mit den gleichen Zusätzen versehen, die auch im PC-DOS möglich sind. Weitere Informationen dazu finden Sie im Kapitel 11.

Achtung:
Verwenden Sie keine Dienstprogramme früherer MS-DOS-Versionen unter der Version 3.10. Es ist dann möglich, daß diese Befehle nicht korrekt ausgeführt werden und wertvolle Daten verlorengehen.

Beachten Sie bitte, daß folgende Zeichen im Dateinamen ab der Version 2.0 nicht mehr erlaubt sind:



Sollten diese Zeichen in einem Dateinamen vorhanden sein, muß diese Datei umbenannt werden, damit sie unter der Version 3.10 bearbeitet werden kann. Das ist unter einer Version kleiner als 2.0 möglich.

Beachten Sie auch die unterschiedlichen Diskettenformate. Disketten der Version 2 und der Version 3 können von der Version 1 nicht gelesen oder verarbeitet werden. Umgekehrt ist es jedoch möglich. Disketten der Version 3 können nicht immer problemlos von der Version 2 bearbeitet werden. Weitere Erklärungen dazu finden Sie auf den Seiten 1.4ff.

Handbuchaufteilung

Allgemeine Informationen

Da dieses Handbuch sicher von Benutzern mit unterschiedlichem Wissensstand gelesen wird, ist es in verschiedene Teile gegliedert. Dadurch wird erreicht, daß jeder Leser mit geringem Aufwand an diejenige Stelle des Handbuches geführt wird, die seinen Kenntnissen entspricht.

Einige Kapitel müssen von allen Benutzern gelesen werden, einige sind sehr hilfreich (aber nicht unbedingt notwendig) und andere sind nur für besondere Hard- oder Softwareanforderungen notwendig.

Jeder Teil dieses Handbuches ist in Kapitel gegliedert. Die einzelnen Seiten dieser Kapitel sind von Beginn eines jeweiligen Kapitels an numeriert. Jedes Kapitel ist wiederum in Abschnitte gegliedert, und einige Abschnitte sind nochmal unterteilt.

Zuerst wird es notwendig sein, sich mit dem Betriebssystem vertraut zu machen. Sie sollten wissen, wofür das Betriebssystem erforderlich ist und was es kann. Danach müssen Sie einige Vorbereitungen treffen, um Ihr Anwenderprogramm starten zu können. Dies wird beschrieben in den Teilen:

- Teil A Einführung
- Teil I Vorbereitung

Einführung

Im Teil A wird der Umfang und das begleitende Handbuch des Betriebssystemes MS-DOS Version 3.10 erklärt. Es wird daher empfohlen, daß alle Benutzer des Zenith PC, die dieses Betriebssystem benutzen, diesen Teil aufmerksam lesen.

Handbuchaufteilung

Vorbereitung

Im Teil I des Handbuches lesen Sie allgemeine Erklärungen. Danach werden Sie schrittweise in das System eingeführt.

Im Kapitel 1 werden Ihnen erste Begriffe des *Betriebssystemes* erklärt. Diese Informationen bilden die Grundlage, auf der alle weiteren Erklärungen aufbauen.

In den Kapiteln 2–4 werden schrittweise die Anwenderprogramme zur späteren Benutzung vorbereitet. Dazu ist es erforderlich, daß Sie nach dem *Systemstart* zuerst Sicherungs- und Arbeitskopien der Programme erstellen. Sie sollten **niemals !!** mit Originaldisketten arbeiten.

Sind bereits Kopien vorhanden, können Sie nach dem Systemstart die Kapitel 3 und 4 überschlagen.

Ergänzende Informationen

Nachdem Sie sich mit Ihrem Zenith PC und dem Betriebssystem vertraut gemacht haben, erhalten Sie in den folgenden Teilen Ihres Handbuches weitere ausführliche Informationen zur MS-DOS Version 3.10:

- Teil II Erste Schritte
- Teil III Befehlserklärungen

Selbst wenn Sie schon Kenntnisse des Betriebssystemes besitzen oder nur mit Anwenderprogrammen arbeiten wollen, erhalten Sie in diesen Kapiteln viele nützliche und wertvolle Hinweise. Diese ermöglichen es Ihnen, im Laufe der Zeit sämtliche Fähigkeiten Ihres Zenith PC voll auszunutzen. Als Ergänzung hierzu ist das kleine Referenzheft beigefügt. Es soll Ihnen bei Ihrer täglichen Arbeit als Gedächnisstütze dienen.

Erste Schritte

Im Teil II werden ausführlich die *Grundzüge* von MS-DOS Version 3.10 beschrieben. Das ist die Befehlseingabe, der Systemstart, das hierarchische Disketteninhaltsverzeichnis, die Informationsübertragung und der Aufbau des Betriebssystemes. Die Erklärungen sind bewußt einfach gehalten, um auch einen Personenkreis anzusprechen, der sich bislang nicht mit Mikrocomputern befaßt hat.

Befehlserklärungen

Im Teil III werden ausführlich die einzelnen *Befehle* von MS-DOS Version 3.10 beschrieben. Der Teil besteht aus zwei Kapiteln, einer *Befehlsübersicht* und einer *Befehlsbeschreibung.* Auch hier werden keine Fachausdrücke zur Erklärung benutzt. In der Übersicht sind die Befehle einmal alphabetisch und einmal nach Funktionen zusammengefaßt aufgelistet. Die Befehlsbeschreibungen erklären in alphabetischer Reihenfolge die einzelnen Befehle. Mit Hilfe von Beispielen wird die richtige Schreibweise erklärt. Zum Schluß der Befehlserklärung erfolgt eine Übersicht möglicher Fehlermeldungen mit Hinweisen zur Behebung.

Befehle für spezielle Anwenderprogramme (Software) und besondere Geräteerweiterungen (Hardware) sind in den folgenden Teilen dieses Handbuches beschrieben.

Spezielle Informationen

Für spezielle Anwenderprogramme und Geräteerweiterungen benötigen Sie zu den bereits beschriebenen Befehlen und Programmen weitere MS-DOS-Dienstprogramme. Wenn Sie selber *Programme entwickeln* wollen oder wenn Sie *Winchesterlaufwerke* an Ihrem Zenith PC anschließen, finden Sie die entsprechenden Programmbeschreibungen in den Teilen:

- Teil IV Weitere Dienstprogramme
- Teil V Dienstprogramme für Winchesterlaufwerke

Handbuchaufteilung

Weitere Dienstprogramme

Im Teil IV werden die MS-DOS-Programme EDLIN, LIB, LINK, EXE2BIN und DEBUG ausführlich beschrieben. Da dieser Teil nur für Benutzer mit Computererfahrung und Programmierkenntnissen gedacht ist, werden hier Fachausdrücke benutzt. Die Beschreibung erfolgt in der Reihenfolge einer möglichen Benutzung dieser Programme.

Dienstprogramme für Winchesterlaufwerke

Im Teil V werden fünf spezielle Dienstprogramme für *Winchesterlaufwerke* ausführlich erklärt. Damit können Sie sämtliche Daten der Winchester löschen, die Winchester neu aufteilen, das Laufwerk für den Systemstart neu festlegen, die Winchester für einen Transport vorbereiten und fehlerhafte Sektoren feststellen. Die Erklärungen erfolgen hier wieder mit einfachen Worten, die kein Wissen über Mikrocomputer voraussetzen.

Es handelt sich hierbei um die Programme PREP, PART, ASGNPART, SHIP und DETECT. Die Erklärung erfolgt in der Reihenfolge der Benutzung.

Die Befehle BACKUP und RESTORE (s. a. Kapitel 11) sind ebenfalls speziell für die Arbeit mit Winchesterlaufwerken vorgesehen.

Gerätehandbücher

Von Lesern dieses Handbuches wird erwartet, daß Sie außer diesem Handbuch weitere Handbücher des Zenith PC besitzen, die Sie zusammen mit dem Gerät erhalten haben. Es sind dies die folgenden Bücher:

- Technisches Handbuch der Zenith-PC-Serie
- Bedienerhandbuch für den Zenith PC

Art der Darstellung

Neben dem erklärenden Text in diesem Handbuch sind auch Bildschirmseiten abgedruckt. Zur besseren Übersicht werden dafür jeweils unterschiedliche Buchstabentypen benutzt. Die Schreibweise (Syntax) der Befehlseingabe wird ebenfalls durch verschiedene Buchstabentypen verdeutlicht.

Textdarstellung

Im erläuternden Text, der mit diesem Buchstabentyp geschrieben ist, wird das momentan erklärte Wort durch *kursive Typen* hervorgehoben. Werden fremde Begriffe zur Erklärung benötigt, wird durch einen in Klammern stehenden Querverweis das erklärende Kapitel dieses Begriffes genannt.

Bildschirmdarstellung

Zu Ihrem besseren Verständnis wird oft der zu erwartende Bildschirm in diesem Handbuch mit abgedruckt. Dies wird durch den folgenden Buchstabentyp angezeigt:

```
                DISKCOPY Version 3.10
Copyright (C) 1985 Zenith Data Systems Corporation
```

Dabei ist möglich, daß der abgedruckte Text nicht immer komplett mit Ihrem Bildschirm übereinstimmt. Durch die Weiterentwicklung der bestehenden Programme erhalten Sie laufend neue Versionen, sowohl der Systemdisketten als auch der Handbuchseiten.

Art der Darstellung

Eingabedarstellung

In diesem Handbuch werden Sie oft aufgefordert, *Eingaben* über die Tastatur (Konsole) vorzunehmen. Diese Eingaben (Befehle, Dateinamen etc.) werden in einer separaten Zeile oder auch im erläuternden Text mit verschiedenen Buchstabentypen hervorgehoben. In der Tabelle i finden Sie eine genaue Aufstellung.

Tabelle i **Erklärung der Eingabezeilen**

PLATZ	BEISPIEL	EINGABEFORM
separate Zeile	GWBASIC	Geben Sie jeden Buchstaben einzeln ein, in diesem Beispiel also sieben.
im Text	**ESC**	Betätigen Sie die bezeichnete Taste.
im Text	**CTRL-BREAK** **CTRL-ALT-DEL**	Betätigen Sie sämtliche so bezeichneten Tasten zugleich.

Art der Darstellung

Tabelle i **Erklärung der Eingabezeilen**

PLATZ	BEISPIEL	EINGABEFORM
eckige Klammern	[ ]	Die Klammern selbst dürfen nicht eingegeben werden. Die Eingabe der Namen und Variablen ist optional und nicht immer erforderlich.
Punkte	...	Die Punkte selber dürfen nicht eingegeben werden. Sie deuten an, daß eine weitere Eingabe von gleichartigen Namen oder Variablen möglich ist.
freie Felder		Es ist die Leertaste zu betätigen (außer wenn durch das freie Feld ein Tastenname **[RETURN]** vom einzugebenden Text getrennt wird).
Satzzeichen	, + - ; :	Es ist das jeweilige Zeichen einzugeben.

Es werden in diesem Buch zur Befehlsdarstellung folgende *amerikanische* Sonderzeichen benutzt:

[ ] \ ¦

Diese Zeichen sind auf einer deutschen Tastatur nach DIN-Norm nicht enthalten. Haben Sie das Programm KEYBGR.COM zur Erzeugung einer deutschen DIN-Tastatur geladen, können Sie diese Zeichen durch die Tastenkombination CTRL-ALT und der entsprechend beschrifteten Taste erzeugen (s. a. Kapitel 11).

Art der Darstellung

Variable

Viele der geforderten Eingaben werden in diesem Handbuch als Variable dargestellt. Ihre Eingabe ist vom jeweiligen Benutzer abhängig. In der folgenden Tabelle werden die benutzten Variablen aufgelistet und erklärt. Eine genaue Erläuterung wird noch einmal im Text vorgenommen.

Tabelle ii **Definition der Variablen**

VARIABLE	ERKLÄRUNG
Dateiname	Name der Datei (ein bis acht Buchstaben) und Zusatz (null bis drei Buchstaben), die durch einen Punkt getrennt werden (z. B. DATEINAM.ZUS) oder Name der Datei ohne Zusatz
.Zus	Namenszusatz, der an den Dateinamen angehängt werden kann, besteht aus einem Punkt und bis zu drei Zeichen
Dateiname1	erster Dateiname in einer Reihe von mehreren
Dateiname2	zweiter Dateiname in einer Reihe (von **Dateiname1** verschieden)
Verzeichnis	Inhaltsverzeichnis, listet die Dateien der Diskette oder des Teilbereiches eines Winchesterlaufwerkes auf
Weg	mehrere Inhaltsverzeichnisse in Folge, die eingegeben werden in der Form: [\ [Verzeichnis [\Verzeichnis [\...]]]]
Dateibez	Dateibezeichnung, besteht aus einer Laufwerkbezeichnung und/oder einem Weg und einem Dateinamen in der Form: [D:] [Weg] Dateiname [D:] [Weg] Dateiname.Zus

Tabelle ii **Erklärung der Variablen**

VARIABLE	ERKLÄRUNG
D:	Laufwerkbezeichnung, sowohl für das Arbeitslaufwerk als auch für das Ziellaufwerk bei Datenübertragung
s:	Ursprungslaufwerk (Source drive) bei Eingaben, die auch ein Ziellaufwerk (Target drive) erfordern
/X	ein oder mehrere zusätzliche Eingaben, die aus einem Schrägstrich und einem einzelnen Buchstaben bestehen; manchmal wird noch eine zusätzliche Eingabe gefordert
p	Nummer des gewünschten Teilbereiches eines Winchesterlaufwerkes
u:	Nummer des angeschlossenen Winchesterlaufwerkes, von 0 bis 7, auf das zugegriffen werden soll
mm	Monat (1 bis 12) oder Minute (0 bis 59)
tt	Tag des Monats (1 bis 31)
jj	Jahr (1980 bis 1999)
jjjj	Jahr (1980 bis 2099)
hh	Stunde (0 bis 59)
ss	Sekunde (0 bis 59)
cc	hundertstel Sekunde
zdn	zusammengefaßte Dateinamen mit Hilfe der Zeichen „*" oder „?"
n	Ziffer mit dezimalem Wert
h	wird einer Ziffer nachgestellt und kennzeichnet damit einen hexadezimalen Wert
Device	Name eines Peripheriegerätes
cttydev	Name des Peripheriegerätes, das Daten empfangen und auch senden kann

Teil I

Vorbereitung

Vorbereitung

In diesem Teil des Handbuches wird erklärt, wie Sie das MS-DOS Betriebssystem vorbereiten müssen, um Ihren Zenith PC in kürzester Zeit starten zu können. Dieser Teil besteht aus den folgenden Kapiteln:

- Kapitel 1 Erste Begriffe
- Kapitel 2 Systemstart
- Kapitel 3 Sicherungskopien
- Kapitel 4 Arbeitskopien

Das Kapitel 1 sollte von allen Benutzern gelesen werden, die noch keine Erfahrung mit Mikrocomputern haben. Es werden allgemeine Begriffe erklärt, die Sie benötigen, um das Betriebssystem MS-DOS zu verstehen. Dieses Kapitel sollte auch von den Benutzern gelesen werden, die zwar Erfahrung mit Mikrocomputer haben, aber bisher noch nicht mit MS-DOS gearbeitet haben.

In den Kapiteln 2 bis 4 werden schrittweise die einzelnen Abläufe erklärt, die Sie nachvollziehen müssen, um Ihre Disketten und Anwenderprogramme für die Benutzung vorzubereiten. Sie haben dann Sicherungskopien Ihrer Disketten erstellt und besitzen ebenfalls Arbeitsdisketten mit dem Betriebssystem MS-DOS.

Falls Sie diese Disketten schon besitzen, können Sie die Kapitel 3 und 4 übergehen.

Kapitel 1

Erste Begriffe

Übersicht

In diesem Kapitel werden Begriffe erklärt, die für das Verständnis des Handbuches wichtig sind. Dies gilt vor allem für Benutzer des Zenith PC, die noch keine Erfahrung mit Mikrocomputern oder Betriebssystemen haben. Es ist ebenso als Überblick für erfahrene Mikrocomputerbesitzer gedacht. Im einzelnen werden in diesem Kapitel folgende Begriffe erklärt:

- Betriebssysteme
- Diskettenplatten
- Winchesterplatten
- Dateien
- Befehle
- Systemstart

Betriebssysteme

Die sichtbaren Teile Ihres Zenith PC bestehen aus einer Zentraleinheit (ZE) und angeschlossenen Peripheriegeräten (Devices). Die Zentraleinheit beinhaltet das Herz Ihres Computers:

den Mikroprozessor (CPU = Central Processing Unit), einen Intel 8088 und den Arbeitsspeicher (RAM = Random Access Memory) mit einem Speichervermögen von maximal 640.000 Zeichen. In der Zentraleinheit sind einzelne Platinen (Karten) vorhanden, die die Koordinierung der angeschlossenen Peripheriegeräte übernehmen. Dieses sind Tastatur, Bildschirm, Disketten- und Winchesterlaufwerke, Drucker und Modems. Die Tastatur und der Bildschirm werden von Ihnen am meisten benutzt werden, da Sie mit Hilfe dieser Peripherien mit dem Computer in Verbindung treten können. Über die Tastatur teilen Sie dem Computer Ihre Befehle mit, auf dem Bildschirm antwortet Ihr Zenith PC dann in aller Regel. Damit der Informationsfluß der verschiedenen Geräte untereinander fehlerfrei und schnell abläuft, benötigen Sie ein eigenes Programm, ein *Betriebssystem.*

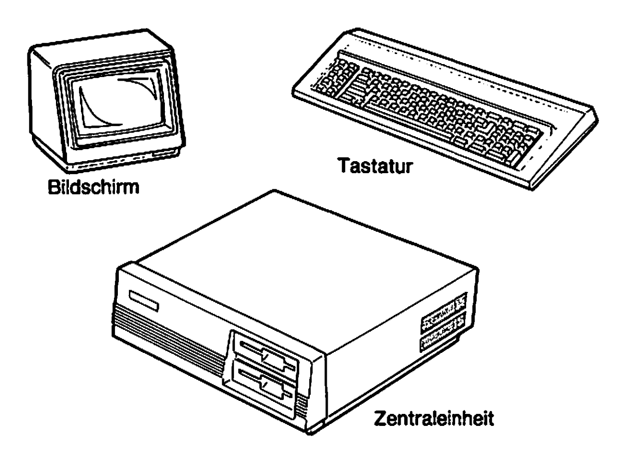


Bild 1.1
Ein Zenith-Mikrocomputer

Betriebssysteme

Das Betriebssystem überwacht (unterstützt) außer den gerade beschriebenen *Geräten* (Hardware) auch die von Ihnen benutzten *Anwenderprogramme* (Software). Ein solches Programm wurde speziell für die Lösung einer einzelnen Aufgabe geschrieben, unabhängig von dem Computer, auf dem es abläuft. Dieses kann beispielsweise ein Textverarbeitungsprogramm (mit einem solchen Programm wurde dieses Handbuch erstellt) oder ein Kalkulationsprogramm sein. Die Verbindung zwischen diesen Programmen und dem Zenith PC wird durch das Betriebssystem hergestellt. Würden die Anwenderprogramme ohne Betriebssystem auf Ihrem Zenith PC ablaufen, wären sie viel umfangreicher und langsamer. So aber kann das Anwenderprogramm bestimmte Aufgaben auf das Betriebssystem übertragen. Dieses wiederum gibt die Befehle an die entsprechenden Peripherieeinheiten weiter, die dann Ihre Befehle ausführen (z. B. Dateien kopieren, drucken). Diese Art der Verbindung zwischen den Programmen wird durch einen *Anruf des Betriebssystemes* (System Call) vom Anwenderprogramm aus hergestellt.

Es ist daher erforderlich, das Betriebssystem von dem Disketten- oder dem Winchesterlaufwerk in den Arbeitsspeicher Ihres Zenith PC zu kopieren (laden), bevor Sie dem Computer eine Aufgabe übertragen. Dies geschieht automatisch sofort nach dem Einschalten. In Ihrem Zenith PC ist ein spezieller Baustein dafür eingebaut, der diesen Ladevorgang durchführt. Es ist ein ROM-Chip (ROM = Read Only Memory, Lesespeicher), und er beinhaltet das „MFM-200 Monitorprogramm“. Der automatische Ladevorgang wird *System-Bootup* genannt. Wie dieses im einzelnen durchgeführt wird, ist im Kapitel 2 beschrieben.

Ist der automatische Ladevorgang erfolgreich beendet, meldet sich das Betriebssystem mit seinem *Anforderungszeichen* (Prompt). Bei MS-DOS ist dies das Zeichen „ > “. Es zeigt an, daß von Ihnen eine Eingabe erwartet und gefordert wird. Zusätzlich wird Ihnen noch das momentane *Arbeitslaufwerk* mitgeteilt. Dies geschieht durch einen Buchstaben (näheres dazu im folgenden Abschnitt). Haben Sie Ihren Zenith PC vom ersten Diskettenlaufwerk aus gestartet, sieht das Anforderungszeichen so aus:

A>

Laufwerke

Ein *Laufwerk* ist ein Peripheriegerät, das Daten zwischen dem Arbeitsspeicher Ihres Computers (interner Speicher) und einem externen Speichermedium überträgt. Dieser externe Speicher kann eine 5.25-Inch-*Diskette* oder aber auch eine 5.25-Inch-*Winchester* sein. Da das Winchesterlaufwerk im Gegensatz zur einzelnen Diskette mehrere nicht biegsame Platten zur Speicherung benutzt, wird es auch *Harddisk* (HD) genannt. Beide Speichermedien arbeiten im Prinzip auf diesselbe Art und Weise. Deshalb wird in diesem Handbuch nur der Begriff „Platte" gebraucht, es sei denn, es wird besonders auf Unterschiede hingewiesen.

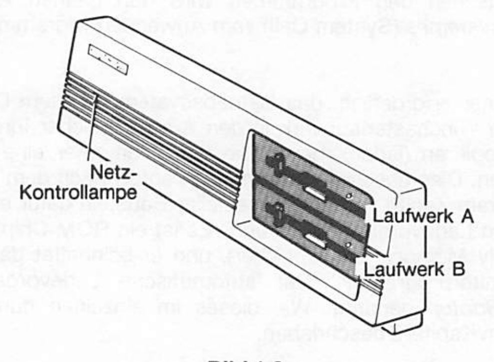


Bild 1.2
Diskettenlaufwerke

In Ihren Zenith PC können Diskettenlaufwerke mit verschiedenen Speicherkapazitäten eingebaut werden. Sie können heute wählen zwischen den folgenden Typen:

- Single-Sided (160K–180K)
- Double-Sided (320K–360K)
- High-Density (1,2M oder mehr)

Auf Grund der unterschiedlichen Aufzeichnungsdichte können Sie nicht Disketten in einem Laufwerktyp bespielen und in einem anderen Laufwerktyp lesen. Es sind nur bestimmte Kombinationen zugelassen:

- Single-Sided-Laufwerke können folgende Disketten lesen und beschreiben:
  Single-Sided-Disketten

- Double-Sided-Laufwerke können folgende Disketten lesen und beschreiben:
  Single-Sided-Disketten
  Double-Sided-Disketten

- High-Density-Laufwerke können folgende Disketten lesen und beschreiben:
  Single-Sided-Disketten
  Double-Sided-Disketten
  High-Density-Disketten

Formatieren Sie eine Diskette in einem Double-Sided-Laufwerk und bespielen diese Diskette in einem High-Density-Laufwerk, können Sie die Diskette nur noch in einem High-Density-Laufwerk benutzen. Achten Sie hierauf vor allem, wenn Ihr Programm mit zwei Diskettenlaufwerken arbeitet.

Laufwerkbezeichnungen

Damit Sie gezielt und schnell auf eine spezielle Platte zugreifen können, werden die verschiedenen, an Ihren Computer angeschlossenen Laufwerke mit den Buchstaben des Alphabetes bezeichnet. Dieses sind die Buchstaben A bis P. Damit von MS-DOS erkannt wird, daß Ihre Eingabe eine *Laufwerkbezeichnung* ist, müssen Sie direkt nach dem jeweiligen Buchstaben einen Doppelpunkt eingeben.

Laufwerke

Ein Beispiel:

A:

Dies ist die Bezeichnung für das erste Diskettenlaufwerk. Bei einem System mit zwei Laufwerken werden diese also mit **A:** für das linke und **B:** für das rechte Laufwerk bezeichnet (s. a. Bild 1.2). Bei nur einem Laufwerk ist dies sowohl A: als auch B: (s. a. Kapitel 5, Seite 26). Winchesterlaufwerke und Teile davon werden mit den Buchstaben C: bis P: bezeichnet. Weitere Informationen dazu im Kapitel 6.

Arbeitslaufwerk

Das momentane *Arbeitslaufwerk* ist ein Laufwerk, auf das vom Betriebssystem bei einer Befehlseingabe zuerst geschrieben oder gelesen (zugegriffen) wird. Sie können dieses verändern, wenn Sie entweder vor der Befehlseingabe die Laufwerkbezeichnung ändern, oder dies bei der Befehlseingabe berücksichtigen. Haben Sie Ihren Zenith PC mit einer Diskette gebootet, so ist das Arbeitslaufwerk A: und wird vom System in folgender Weise angezeigt:

```
A>
```

Haben Sie Ihren Zenith PC vom Winchesterlaufwerk gebootet, so ist das Arbeitslaufwerk C: und wird so angezeigt:

```
C>
```

Möchten Sie jetzt das Laufwerk wechseln, um z.B. auf das zweite Diskettenlaufwerk zu gelangen, geben Sie **B:** ein und betätigen die Taste **RETURN**. Das Systemanforderungszeichen ändert sich jetzt und sieht so aus:

```
B>
```

Jeder Buchstabe, den Sie eingeben, muß ein tatsächlich vorhandenes Laufwerk bezeichnen. Die Eingabe darf mit einem großen oder kleinen Buchstaben erfolgen. Die bezeichneten Laufwerke können Diskettenlaufwerke oder Winchesterlaufwerke oder auch Teile der

Winchesterlaufwerke sein. Es ist nicht erforderlich, daß sich auf diesen Laufwerken ein Betriebssystem befindet. Dieses ist ja bereits schon im Arbeitsspeicher. Geben Sie eine Bezeichnung für ein nicht vorhandenes Laufwerk ein, erscheint folgende Fehlermeldung:

Invalid drive specification	Falsche Laufwerkbezeichnung

Obwohl Disketten- und Winchesterplatten aus verschiedenem Material sind, ist die Art und Weise, wie Informationen gelesen und geschrieben werden, bei beiden gleich. Die Daten oder Informationen werden in konzentrischen Ringen auf der Oberfläche gespeichert. Diese Ringe werden *Spuren* (Track) genannt. Die Spuren sind unterteilt in einzelne *Bereiche* (Sektoren). Die Speicherkapazität dieser Spuren und Bereiche wird in *Byte* gemessen. Jedes Byte wiederum besteht aus *8 Bit* (1 Bit ist die kleinste Informationseinheit im Computer). Ein einzelner Buchstabe oder Befehl, den Sie eingeben, wird als Byte gespeichert. Da Byte eine sehr kleine Maßeinheit ist, wird die Speicherkapazität meistens in *Kilobytes* (KB) angegeben. Dabei ist 1 Kilobyte gleich 1024 Bytes und 128 Kilobytes sind 131 072 Bytes.

Informationen werden in Form von elektrischen Impulsen übertragen. Ein Elektromagnet im Laufwerk schreibt oder liest diese Informationen auf der Platte. Dieser Magnet wird daher auch *Schreib-Lese-Kopf* genannt. Der Schreib-Lese-Kopf wird vom Betriebssystem kontrolliert und kann zu jeder beliebigen Stelle der Platte dirigiert werden. Da diese Platte mit einer sehr hohen Geschwindigkeit (Diskette 300 U/sec.; Winchester 3000 U/sec.) läuft, werden die Daten sehr schnell übertragen. Der Schreib-Lese-Kopf sucht sich selber die geforderte Stelle auf der Platte und liest oder schreibt die gewünschten Daten. Diese Informationen erhält er von einem *Inhaltsverzeichnis*, das auf jeder Platte vorhanden ist. In diesem Verzeichnis stehen die Namen der gespeicherten Dateien mit den dazugehörigen Positionen auf der Platte.

Diskettenplatten

Eine *Diskettenplatte* besteht aus einer 5.25 Inch großen runden Plastikscheibe, überzogen mit einer feinen magnetisierbaren Oxidschicht. Diese Scheibe wird in einer Hülle aufbewahrt, die sie fast vollständig bedeckt. Nur ein kleiner schmaler Schlitz für

den Schreib-Lese-Kopf bleibt frei, der Schreib-Lese-Schlitz. Die Scheibe selber und auch die darauf gespeicherten Informationen sind sehr leicht zu zerstören. Sie sollten die Disketten daher immer sehr sorgfältig behandeln und folgendes beachten:

- Wenn Sie die Diskette aus ihrer Verpackung nehmen, berühren Sie niemals die Oberfläche, vor allem nicht am Schreib-Lese-Schlitz.
- Ist die Diskette nicht im Laufwerk, so stecken Sie sie wieder in die Verpackung.
- Öffnen Sie die Klappe des Laufwerkes erst, wenn die Kontrolllampe wieder erloschen ist.
- Achten Sie darauf, daß kein Staub oder irgendwelche Flüßigkeiten auf die Scheibe gelangen.
- Bewahren Sie die Diskette nicht in der Nähe magnetischer Felder auf. Selbst ein Kugelschreiber kann schon Daten zerstören.
- Gehen Sie mit Disketten nicht durch Röntgenanlagen (z. B. am Flughafen).
- Lassen Sie keine Diskette im Laufwerk, wenn Sie den Strom der ZE ein- oder ausschalten (Ausnahme: Systemstart). Durch eventuelle Stromstöße kann der Schreib-Lese-Kopf auf die *Platte stürzen* (Head-Crash).

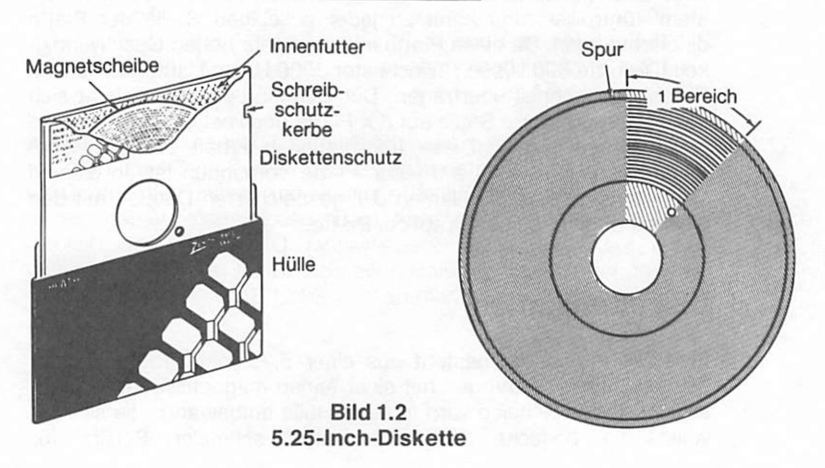


Bild 1.2
5.25-Inch-Diskette

- Achten Sie darauf, daß die Disketten nicht zu heiß (über 52 Grad) oder zu kalt (unter 10 Grad) gelagert werden.

- Benutzen Sie zum Beschriften der Etiketten niemals einen Kugelschreiber, sondern immer einen weichen Filzschreiber.

- Achten Sie darauf, daß Disketten niemals gebogen oder gestapelt werden.

Für die verschiedenen Laufwerktypen werden auch Disketten mit verschiedenen Speicherkapazitäten hergestellt. Dabei können Sie eine höherwertige Diskette immer in einem Laufwerk mit niedrigerer Speicherkapazität formatieren, aber niemals umgekehrt. Es gibt folgende Diskettentypen:

- Single-Sided-Disketten mit
  40 tpi
  8 oder 9 Sektoren pro Spur
  160–180K Speichervermögen

- Double-Sided-Disketten mit
  40 tpi
  8 oder 9 Sektoren pro Spur
  320–360K Speichervermögen

- High-Density-Disketten mit
  80 tpi
  15 Sektoren pro Spur
  1,2–1,6M Speichervermögen

Wollen Sie sich gegen unbeabsichtigtes Löschen von Daten oder Überschreiben von Daten schützen, so verdecken Sie die Kerbe an der linken Seite der Diskette mit einem kleinen Streifen. Diese Disketten können dann nicht mehr beschrieben werden, es sei denn, Sie entfernen die Sperre wieder. Die Daten dieser Diskette können jetzt nur noch gelesen werden. Es ist jetzt eine *schreibgeschützte* (write protect) Diskette (s. a. Bild 1.3).

Sollten Sie dennoch versuchen auf diese Diskette zu schreiben, erscheint folgende Meldung:

Write protect error writing drive X, Abort, Retry, Ignore?	Schreibfehler in Laufwerk X Abbrechen, Wiederholen, Ignorieren?

Sie können den Vorgang jetzt abbrechen oder die Schreibsperre entfernen und wiederholen.

Mit dem Dienstprogramm ATTRIB können Sie eine oder mehrere Dateien ebenfalls schützen. Das Programm markiert diese Dateien als Nur-Lese-Dateien und schützt sie somit softwaremäßig.

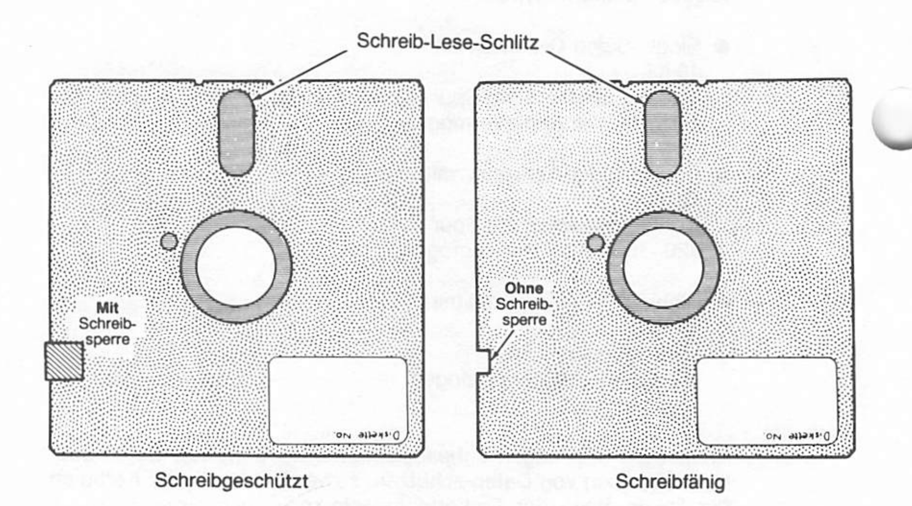


Bild 1.3
Schreibschutz der Diskette

Disketten werden mit dem Schreib-Lese-Schlitz zuerst in das Laufwerk hineingeschoben. Die Kerbe für den Schreibschutz befindet sich dabei auf der linken Seite. Erst wenn die Diskette richtig im Laufwerk ist, kann die Klappe geschlossen werden.

High-Density-Laufwerke

Ist Ihr Zenith PC mit High-Density-Laufwerken ausgerüstet, können Sie auf einer Diskette mehr als 1 Megabyte an Daten speichern. Dies ist ein vielfaches der bisherigen Laufwerke. Damit benötigen Sie für viele Programme nur noch eine Diskette. Auch können Sie das hierarchische Inhaltsverzeichnis benutzen. Sie sollen jedoch einige Regeln bei der Benutzung beachten:

- Ist Ihr Gerät nur mit High-Density-Laufwerken ausgerüstet, können Sie beliebig Disketten bespielen und lesen. Ebenso können Sie alle Disketten lesen, die von PC unter MS-DOS formatiert wurden. Ihre Disketten können jedoch nicht von allen anderen Geräten gelesen werden, verfügen diese Geräte nur über herkömmliche Laufwerke.

- Ist Ihr Gerät mit einem High-Density-Laufwerk und einem herkömmlichen Laufwerk ausgerüstet, können Sie die Disketten nicht mehr beliebig tauschen. Disketten des High-Density-Laufwerkes können nicht mehr vom herkömmlichen Laufwerk gelesen werden. Haben Sie in diesem eine Diskette formatiert, und bespielen Sie diese Diskette in einem High-Density-Laufwerk, kann die Diskette nicht mehr vom herkömmlichen Laufwerk gelesen werden.

- Benutzen Sie ein Anwendungsprogramm auf einem Gerät mit einem High-Density-Laufwerk und einem herkömmlichen Laufwerk, sollten Sie die Disketten nicht tauschen. Ist dies erforderlich, versehen Sie die Diskette des herkömmlichen Laufwerkes mit einem Schreibschutz. Sonst können Sie die Diskette nur noch im High-Density-Laufwerk benutzen.

Achten Sie ebenfalls darauf, daß Sie die Disketten des Formates HD/DS nicht in einem herkömmlichen Laufwerk benutzen.

Winchesterplatten

Ein *Winchesterlaufwerk* besteht aus mehreren harten, nicht biegsamen Metallscheiben, die von einer feinen magnetisierbaren Oxidschicht bedeckt sind. Diese Platten sind in einem staubfreien, luftdicht verschlossenen Gehäuse untergebracht. Dieses Gehäuse kann sowohl in die ZE integriert, als auch separat aufgestellt, werden. Die Speicherkapazität ist sehr viel größer, als die einer Diskette. Ebenso ist die Zugriffszeit wegen der höheren Rotationsgeschwindigkeit kürzer. Um die hohe Speicherkapazität des Winchesterlaufwerkes voll auszunutzen, sollten Sie Ihre Dateien in verschiedenen Inhaltsverzeichnissen (s. a. Kapitel 7) speichern oder das Laufwerk in verschiedene Teilbereiche aufteilen (s. a. Kapitel 4).

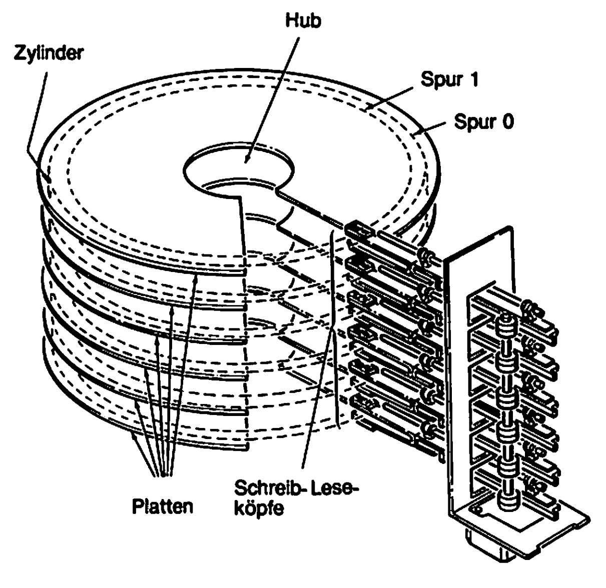


Bild 1.4
Winchesterlaufwerk

Winchester- und Diskettenlaufwerke werden vielfach gleich behandelt:

- auf Programme oder Daten wird in gleicher Weise zugegriffen;
- jedes Laufwerk kann ein eigenes Betriebssystem enthalten, das beim Start geladen werden kann;
- jedes Laufwerk kann mit den verschiedenen Formatprogrammen zur Datenaufnahme vorbereitet werden.

Winchester- und Diskettenlaufwerke unterscheiden sich in den folgenden Punkten:

- die Speicherkapazität eines Winchesterlaufwerkes ist sehr viel größer als die eines Diskettenlaufwerkes;
- die Speicherkapazität eines Diskettenlaufwerkes ist begrenzt durch seine physikalische Größe; die Speicherkapazität eines Winchesterlaufwerkes kann durch verschiedene Dienstprogramme in weiten Bereichen variiert werden;
- Diskettenplatten können leicht separat transportiert werden; Winchesterplatten können nicht aus dem Gehäuse herausgenommen werden;
- nach dem Systemstart kann sofort auf alle Diskettenlaufwerke zugegriffen werden; nach dem Systemstart kann nur auf einen Teil des Winchesterlaufwerkes zugegriffen werden, die anderen Teile müssen erst mit dem Befehl ASGNPART bestimmt werden.

Dateien

Das Betriebssystem MS-DOS erlaubt es Ihnen, Informationen zu erzeugen, zu speichern, zu verändern und zu analysieren. Dazu werden diese Informationen in Gruppen gespeichert, die *Dateien* genannt werden. Diese Dateien können sowohl Programme als auch durch Programme erfaßte Daten (z. B. Adressen) beinhalten. Diese Dateien erhalten von Ihnen einen Namen. Geben Sie diesen

Dateien

Namen ein, wird er vom Betriebssystem analysiert. Der Schreib-Lese-Kopf sucht dann automatisch auf dem angegebenen Laufwerk die so benannten Daten. Diese werden dann, falls tatsächlich vorhanden, auf dem Bildschirm angezeigt. Ebenso werden diese Daten, falls erforderlich, durch das Betriebssystem kopiert, bevor sie weiterverarbeitet werden. Die Dateinamen müßen bestimmten Regeln entsprechen, damit sie von MS-DOS verarbeitet werden können.

Dateiname und Dateinamenzusatz

Eine Datei wird durch einen Namen bestimmt, der aus zwei Teilen besteht: dem *Dateinamen* und dem *Dateinamenzusatz*. Der Dateiname besteht aus maximal acht Buchstaben, der Dateinamenzusatz besteht aus maximal drei Buchstaben; beide werden durch einen Punkt miteinander verbunden. Ein Beispiel:

DATEINAM.ZUS

Die Eingabe des Punktes und des Zusatzes ist nicht erforderlich. Besteht ein Dateiname aus weniger als acht Buchstaben, werden automatisch entsprechende Leerfelder angehängt; ebenso beim Dateinamenzusatz. Ein Beispiel:

DATEI.Z

Im Dateinamen und im Zusatz dürfen folgende Buchstaben, Ziffern und Zeichen der Tastatur verwendet werden:

A–Z	0–9	$	&	#
'	(	)	-	
%	!	-	^	

Die Eingabe darf mit Groß- oder Kleinbuchstaben erfolgen. Kleinbuchstaben werden bei der Eingabe automatisch in Großbuchstaben umgeformt. Ein Beispiel:

DateiNam.Zus

wird als: *DATEINAM.ZUS* gespeichert.

Im Dateinamen sind Kontrollzeichen und folgende Zeichen nicht erlaubt:

? . , ; : =

* / \ + " <

> ¦

Diese Zeichen sind für besondere Funktionen reserviert.

Die folgenden Beispiele zeigen gültige Dateinamen:

FORMAT.COM	memo.doc	IO.SYS
4-7-83.TXT	JOB3.HEX	FILE#1
33%-RATE.DAT	ZMSGS.OVR	

Dateinamenzusätze sind nicht unbedingt erforderlich, aber sehr hilfreich, da sie vielfach anzeigen, um welche Datei es sich handelt. Die folgende Auflistung gibt einen kleinen Überblick:

Tabelle 1.1 Dateinamenzusätze

ZUSATZ	ERKLÄRUNG
ASM	ASSEMBLER - Ursprungsdatei
BAK	Sicherungsdatei
BAS	BASIC - Ursprungsdatei
BAT	Datei für automatische Befehlsverarbeitung
BIN	Binärdatei
COB	COBOL - Ursprungsdatei
COM	ausführbare Datei unter MS-DOS
DAT	ASCII - Datei
DIF	Datei des Dienstprogrammes FC
DOC	ASCII - Datei
EXE	ausführbare Datei unter MS-DOS
FOR	FORTRAN - Ursprungsdatei
LST	ASCII - Liste
OBJ	Objektdatei, erzeugt durch einen Compiler
SYS	Systemdatei
$$$	Arbeitsdatei

Dateibezeichnung

Dem einzugebenden Dateinamen können Sie eine Laufwerkbezeichnung oder auch den oder die Namen von Inhaltsverzeichnissen voranstellen. Dies ist immer dann erforderlich, wenn die Datei nicht auf dem momentanen Arbeitslaufwerk gespeichert ist. Sie arbeiten auf dem Laufwerk C: und wollen die Datei „TEST.DAT" von der Diskette überspielen. Dann geben Sie ein:

```
COPY A:TEST.DAT
```

Sind die Dateien auf den Laufwerken in unterschiedlichen Inhaltsverzeichnissen gespeichert, so können Sie die Namen dieser Inhaltsverzeichnisse nach der Laufwerkbezeichnung und vor dem Dateinamen mit eingeben. MS-DOS sucht dann auf dem Laufwerk im angegebenen Verzeichnis. Ein Beispiel:

```
COPY A:TEXTTEST.DAT
```

Jetzt wird die Datei „TEST.DAT", die sich auf dem Laufwerk A: befindet, kopiert. Da die Datei in einem Unterverzeichnis gespeichert ist, wird der Name des Unterverzeichnisses „TEXT" mit eingegeben.

Die zusätzliche Eingabe eines Laufwerkes oder auch eines Inhaltsverzeichnisses vor dem Dateinamen wird *Dateibezeichnung* genannt.

Reservierte Dateinamen für Peripheriegeräte

Verschiedene Namen sind für MS-DOS reserviert. Diese Namen dürfen von Ihnen nicht zur Benennung Ihrer Dateien benutzt werden. Mit diesen Namen steuert MS-DOS die Ein- und Ausgabe der Peripheriegeräte. Benutzen Sie diese Namen dennoch, rufen Sie nicht Ihre Datei auf, sondern das Peripheriegerät. Die reservierten Namen sind:

- AUX = asynchrone Schnittstelle
- CLOCK$ = Treiber für Echtzeit-Uhr
- COM1 = erster Seriellausgang
- COM2 = zweiter Seriellausgang
- CON = Console
- LPT1 = erster Parallelausgang
- LPT2 = zweiter Parallelausgang
- LPT3 = dritter Parallelausgang
- NUL = nicht vorhandener Ausgang
- PRN = Drucker

Zeichen zur Zusammenfassung mehrerer Dateien

Die meisten der eingegebenen Befehle beziehen sich auf eine genau spezifizierte Datei. Manchmal ist es aber wünschenswert, mehrere gleichartige Dateien aufzurufen. Sie können jede dieser gleichartigen Dateien nacheinander aufrufen, indem Sie den gleichen Befehl mehrmals eingeben. Sie können aber auch mit nur einem Befehl alle gleichartigen Dateien zusammen aufrufen. Dafür geben Sie nur die übereinstimmenden Teile der Dateinamen oder Dateibezeichnungen ein. Für den restlichen Teil des Dateinamens verwenden Sie *besondere Zeichen*, die von MS-DOS als Variable erkannt werden und den Dateinamen dann vervollständigen.

Diese Variable sind das *Fragezeichen* (?) und der *Stern* (*).

Das Fragezeichen kann für einen einzelnen Buchstaben im Dateinamen oder Namenszusatz stehen. Es dürfen mehrere Fragezeichen in einem Dateinamen mit eingegeben werden. Ein Beispiel:

```
DIR TEST?DAT.TXT
```

zeigt alle gespeicherten Dateien an, die bis auf den fünften Buchstaben des Dateinamens übereinstimmen. Dieses könnten folgende Dateien sein:

TEST1DAT.TXT
TEST2DAT.TXT
TESTADAT.TXT

Das Fragezeichen darf an jeder beliebigen Stelle des Namens und des Zusatzes benutzt werden.

Im Gegensatz dazu kann der Stern für mehrere verschiedene Zeichen benutzt werden. Er kann für einen oder aber auch für alle Zeichen des Dateinamens und Zusatzes benutzt werden. Ab der Stelle, an der er eingegeben wird, ersetzt er alle Zeichen.

Zeichen zur Zusammenfassung mehrerer Dateien

Ein Beispiel:

DIR TEST*.TXT

zeigt alle Dateien mit folgenden Namen an:

TEST1DAT.TXT
TEST2DAT.TXT
TESTADAT.TXT
TESTNEU1.TXT
TESTNEU2.TXT

Es ist aber auch möglich, alle vorhandenen Dateien eines Inhaltsverzeichnisses mit Hilfe des Sterns anzuzeigen. Ein Beispiel:

DIR *.*

Hier stehen die Sterne für sämtliche Zeichen des gesamten Dateinamens und des Namenszusatzes. Da der Stern ab der Stelle, wo er eingesetzt wird, alle Zeichen ersetzt, ist eine Eingabe wie diese nicht erlaubt: DIR *DAT.TXT. Erlaubt ist aber, den Stern nur für den Dateinamen oder den Zusatz einzugeben. Ein Beispiel:

DIR TEST.*

Jetzt werden alle Dateien aufgelistet, die mit „TEST" bezeichnet wurden und sich nur im Zusatz unterscheiden.

Diese Zeichen zur Zusammenfassung mehrerer Dateien werden meistens in Verbindung mit den Befehlen COPY und DIR benutzt.

Dateisicherung

Damit Sie Ihren Zenith PC immer so nutzen können, wie Sie es sich wünschen, sollten Sie von allen Programmen und Dateien *Sicherungskopien* anfertigen. Dieses beginnt bereits, wenn Sie ein neues Programm bekommen oder eine neue Version eines Programmes. Das gilt ebenso für Daten, die Sie bei Ihrer täglichen Arbeit erfassen und verarbeiten.

Arbeiten Sie daher nie mit Originaldisketten, sondern fertigen Sie immer erst eine Kopie der neuen Diskette an. Diese Kopie sollten Sie dann benutzen, um davon Ihre Arbeitskopie zu erstellen. Verwahren Sie die Kopien immer getrennt von Ihren Originalen auf.

Zeichen zur Zusammenfassung mehrerer Dateien

Verfahren Sie genauso mit Ihren individuellen Daten. Um diese Daten zu sichern, sollten Sie in regelmäßigen, nicht zu großen, Abständen Sicherungskopien (Backup) anfertigen. Die Daten des Winchesterlaufwerkes lassen sich sehr einfach mit dem Befehl BACKUP (s. a. Kapitel 11) kopieren. So ersparen Sie sich unter Umständen hohe Kosten für die Wiederherstellung verlorengegangener Daten. Mit dem Programm ATTRIB können Sie Dateien einen Softwareschutz geben. Damit können Sie die Dateien nur lesen, aber nich mehr verändern, auch nicht den Dateinamen.

Befehle

Befehlsarten

Befehle sind Anweisungen an das Betriebssystem, bestimmte Aufgaben durchzuführen. Diese können sich auf Datenverarbeitung, Datentransfer oder Datenanalyse beziehen. Die meisten Befehle werden von Ihnen über die Tastatur (Konsole) eingegeben. Sie können aber auch in einer speziellen Datei gespeichert werden und rufen sich dann selber auf. Näheres dazu im Kapitel 5.

Das Betriebssystem MS-DOS unterscheidet zwei Arten von Befehlen:

- Interne Befehle
- Externe Befehle

Interne Befehle

Interne Befehle sind jederzeit aufrufbar. Sie werden durch die Datei COMMAND.COM verarbeitet. Diese Datei wird beim Systemstart geladen und von keinem Programm überschrieben oder gelöscht.

Externe Befehle

Externe Befehle sind eigene kleine Dienstprogramme, die vor jedem Aufruf in den Arbeitsspeicher geladen werden müssen. Diese Programme werden von nachfolgenden Programmen dann wieder überschrieben und gelöscht. Damit sie aufgerufen werden können, müssen sie auf der Arbeitsdiskette gespeichert sein. Ist das nicht der Fall, erscheint die Fehlermeldung:

Bad command or filename	Befehl oder Datei falsch

Bei intensivem Einsatz des Betriebssystemes zur Unterstützung der Anwenderprogramme ist es von Vorteil, die am meisten benutzten Dienstprogramme auf die jeweilige Arbeitsdiskette zu kopieren. Ist Ihr Zenith PC mit einem Winchesterlaufwerk ausgerüstet, können Sie auch sämtliche MS-DOS-Dienstprogramme in einem eigenen Inhaltsverzeichnis abspeichern (s. a. Kapitel 7).

Befehlszeile

Die von Ihnen geforderte oder erwartete Eingabe, wenn sich das Betriebssystem mit seinem Anforderungszeichen (Prompt) meldet, wird in die *Befehlszeile* geschrieben. Hier sehen Sie nach dem Prompt eine kleine *blinkende Linie*. Dies ist der Cursor. Er bewegt sich bei jeder Eingabe, die Sie machen, mit. Er zeigt Ihnen somit an, wo Ihre Eingabe auf dem Bildschirm erscheint. Die erste Eingabe in dieser Zeile ist immer der Name eines Befehls. Sie haben vielfach die Möglichkeit, nach dem Befehlsnamen ergänzende Angaben mit einzugeben. Damit werden verschiedene Möglichkeiten der Befehlsausführung gewählt oder die zu bearbeitende Datei bestimmt. Die Eingabe sieht folgendermaßen aus:

```
BEFEHL Ergänzende Angaben
```

Die meisten Befehle werden von Ihnen über die Tastatur eingegeben. Jede Eingabe in die Befehlszeile muß mit Betätigen der Taste **RETURN** abgeschlossen werden. Die folgenden Informationen gelten für alle Befehle:

- Das Systemanforderungszeichen, bestehend aus dem Buchstaben des Arbeitslaufwerkes und der Klammer, zeigt an, daß eine Eingabe erwartet wird.
- Die meisten eingegebenen Befehle enthalten weitere ergänzende Angaben.
- Die Eingabe darf mit Groß- und Kleinbuchstaben erfolgen.
- Befehle und ergänzende Angaben müssen durch Abgrenzungszeichen getrennt werden. Dies sind Leertaste, Komma (,), Semikolon (;), Gleichheitszeichen (=) und Tabulatortaste.
- Für die Dateibezeichnung (D:DATEINAM.ZUS) dürfen nur Doppelpunkt (:) und Punkt (.) als Abgrenzungszeichen benutzt werden.
- Enthält der einzugebende Dateiname einen Zusatz, muß dieser Zusatz mit eingegeben werden.

Befehle

- Laufwerkbezeichnungen werden vielfach als Ursprungslaufwerk (Source Drive) oder als Ziellaufwerk (Target Drive) beschrieben. Dabei werden die Daten vom Ursprungslaufwerk zum Ziellaufwerk übertragen.
- Zeichen zur Zusammenfassung mehrerer Dateien und reservierte Dateinamen dürfen nicht im Befehlsnamen gebraucht werden.
- Fehler in der Befehlszeile können mit Hilfe der Funktionstasten korrigiert werden.
- Jede Befehlszeile muß mit Betätigen der Taste **RETURN** abgeschlossen werden.
- Jede Befehlsausführung kann durch Betätigen der Tastenkombination **CTRL-BREAK** oder **CTRL-C** unterbrochen werden.
- Bei Bildschirmausgaben, die sich über mehr als 24 Zeilen erstrecken, werden die auszugebenden Zeilen nach oben gerollt. Dieser Vorgang kann durch Betätigen der Tastenkombination **CTRL-NUM LCK** oder **CTRL-S** unterbrochen werden. Die Ausgabe wird fortgesetzt, sobald eine beliebige Taste betätigt wird.
- Werden Sie vom System aufgefordert, zur Fortsetzung des Programms eine Taste zu betätigen (Press any key when ready), dann kann dies irgendeine alphanumerische Taste sein.

Tastenbefehle

Einige Funktionen des Betriebssystemes lassen sich durch Betätigen einer *einzelnen Taste* oder einer *Tastenfolge* aufrufen. Die meisten dieser Funktionen werden durch eine Kombination der Taste **CTRL** mit einer weiteren Taste aufgerufen. Dafür müssen Sie die Taste **CTRL** betätigen und festhalten. Danach betätigen Sie eine weitere Taste. Anschließend lassen Sie alle Tasten gleichzeitig los. Zwei Beispiele:

CTRL-BREAK bricht die Abarbeitung eines Programmes sofort ab

CTRL-ALT-DEL startet (bootet) das System neu

Eine Übersicht sämtlicher Tastenbefehle finden Sie im Kapitel 5.

Erster Systemstart

Bevor Sie das erste Mal Ihren neuen Zenith PC in Betrieb nehmen, müssen Sie folgende Schritte sorgfältig durchführen:

1. Betriebssystem laden

2. Sicherungskopie der Systemdiskette erstellen

3. Arbeitsdiskette anfertigen

Diese Schritte werden in den Kapiteln 2 bis 4 ausführlich beschrieben. Der unerfahrene Mikrocomputeranwender sollte diese Kapitel sorgfältig durchlesen, bevor er den Zenith PC einschaltet.

Zusammenfassung

Das Betriebssystem stellt die Verbindung zwischem dem Computer und den Anwenderprogrammen her. Es ermöglicht Ihnen, Ihren Zenith PC optimal für bestimmte Aufgaben einzusetzen.

Laufwerke werden durch einzelne Buchstaben benannt. Dadurch haben Sie die Möglichkeit, gezielt und schnell auf ein bestimmtes Laufwerk zuzugreifen (lesen oder schreiben). Das Arbeitslaufwerk ist das Laufwerk, auf welches das Betriebssystem zugreift, wenn kein spezielles Laufwerk im Befehl mit angegeben wird.

Es gibt zwei verschiedene externe Speicher: eine 5.25-Inch-Diskette und eine 5.25-Inch-Winchester.

Sie können Ihr Gerät mit drei verschiedenen Laufwerktypen für Disketten ausrüsten. Single-Sided-Laufwerke und entsprechende Disketten können maximal 180K speichern. Double-Sided-Laufwerke können maximal 360K speichern, und High-Density-Laufwerke können 1,2M speichern. Disketten der High-Density-Laufwerke können Sie nicht von den anderen Laufwerktypen lesen lassen.

Die gespeicherten Informationen (Daten oder Programme) werden vom System durch Dateinamen erkannt. Ein Dateiname besteht aus bis zu acht Zeichen und wahlweise einem Zusatz aus bis zu drei Zeichen. Der Name und der Zusatz werden durch einen Punkt getrennt. Bei Eingabe des Namens kann eine Laufwerkbezeichnung und/oder der Name eines Inhaltsverzeichnisses vorangestellt werden. Dies ist dann eine Dateibezeichnung.

Dem Betriebssystem werden Arbeitsanweisungen in Form von Befehlen gegeben. Das System fordert diese durch ein besonderes Anforderungszeichen (Prompt) an. MS-DOS kennt interne und externe Befehle. Interne Befehle sind jederzeit aufrufbar. Externe Befehle sind eigene Programme, die jedesmal neu in den Arbeitsspeicher geladen werden müssen. Einige Befehle können auch durch Betätigen einzelner Tasten eingegeben werden.

Bevor Sie mit Ihrem Computer arbeiten können, müssen Sie das System starten (booten), eine Sicherungskopie der Systemdisketten erstellen und eine Arbeitsdiskette anfertigen.

Kapitel 2

Systemstart

Übersicht

In diesem Kapitel wird Ihnen erklärt, wie Sie das Betriebssystem MD-DOS das erste Mal starten. Haben Sie das Kapitel richtig durchgearbeitet, erscheint das Anforderungszeichen (Prompt) in der Befehlszeile, und Sie können den ersten Befehl eingeben.

Außer Ihrem Zenith PC benötigen Sie dafür die mitgelieferten Originalsystemdisketten MS-DOS Version 1.25 oder Version 3.10.

Es werden in diesem Kapitel die folgenden Schritte erklärt:

- Einlegen der Diskette
- System laden
- Datumseingabe
- Zeiteingabe

Dieses bezieht sich auf ein neues Gerät, daß noch nicht modifiziert wurde. Der Zenith PC wird werkseitig auf automatischen Systemstart eingestellt. Je nach Ausrüstung wird daher von Diskette oder Winchester gestartet. Wird das System von Ihnen modifiziert, kann der Startvorgang anders ablaufen.

Befehlsablauf

Schritt 1: Legen Sie die Systemdiskette eins (für Version 3.10) in das linke Diskettenlaufwerk (Laufwerk A:), wie auf Seite 1.8 beschrieben.

Ist Ihr Gerät mit einem oder zwei Diskettenlaufwerken ausgerüstet, fahren Sie bitte mit Schritt 4 fort.

Ist Ihr Gerät mit einem Winchesterlaufwerk ausgerüstet, fahren Sie bitte mit Schritt 2 fort.

Schritt 2: Ist das Gerät ausgeschaltet, schalten Sie erst alle angeschlossenen Peripheriegeräte ein und danach erst die Zentraleinheit. Betätigen Sie dann die Taste **ESC**, um den automatischen Ladevorgang vom Winchesterlaufwerk abzubrechen. Fahren Sie bitte mit Schritt 3 fort.

Ist Ihr Gerät eingeschaltet, betätigen Sie die drei Tasten **CTRL, ALT** und **DEL** gleichzeitig. Lassen Sie alle Tasten auch gleichzeitig wieder los. Das System startet dann. Betätigen Sie dann die Taste **ESC**, um den automatischen Ladevorgang vom Winchesterlaufwerk abzubrechen. Fahren Sie bitte mit Schritt 3 fort.

Schritt 3: Es erscheint dann in der linken oberen Ecke Ihres Bildschirmes die Statusmeldung des Monitorprogrammes. Das Anforderungszeichen zur Befehlseingabe ist ein Pfeil: „-->“. Die Meldung sieht folgendermaßen aus:

```
MFM-200 Monitor, Version 0.8
Memory Size: 512K bytes
Enter "?" for help.
-->
```

In der ersten Zeile steht der Name des Programmes, sowie die Versionsnummer. In der zweiten Zeile wird der verfügbare Arbeitsspeicher angezeigt. In der dritten Zeile erscheint als Gedächnisstütze das Fragezeichen, welches Sie eingeben können, um eine Auswahl der Eingabemöglichkeiten des Monitorprogrammes zu erhalten. Nähere Einzelheiten dazu finden Sie im Gerätehandbuch. In der vierten Zeile sehen Sie das Anforderungszeichen für Ihre Eingabe.

Diese Statusmeldung erscheint immer dann, wenn Sie Ihr System neu starten und kein automatischer Ladevorgang erfolgt. Sie erscheint ebenfalls, wenn der Startvorgang unterbrochen wird oder keine Systemdiskette im Laufwerk liegt.

Geben Sie jetzt ein **B** (Boot) und ein **F** (Floppy) ein, und betätigen Sie die Taste **RETURN**. Es wird jetzt vom Diskettenlaufwerk A: aus gestartet. Fahren Sie bitte mit Schritt 5 fort.

Schritt 4: Ist das Gerät ausgeschaltet, schalten Sie erst alle angeschlossenen Peripheriegeräte ein und danach erst die Zentraleinheit. Jetzt wird das System automatisch von der Diskette im Laufwerk A: geladen. Fahren Sie bitte mit Schritt 5 fort.

Ist Ihr Gerät eingeschaltet, betätigen Sie die drei Tasten **CTRL, ALT** und **DEL** gleichzeitig. Lassen Sie alle Tasten auch gleichzeitig wieder los. Jetzt wird das System automatisch von der Diskette im Laufwerk A: geladen. Fahren Sie bitte mit Schritt 5 fort.

Ist Ihr System modifiziert worden, so daß kein automatischer Startvorgang erfolgt, erscheint die Statusmeldung des Monitorprogrammes wie im Schritt 3 beschrieben. Geben Sie dann ein **B** (Boot) ein und betätigen Sie die Taste **RETURN**, um den Startvorgang einzuleiten. Fahren Sie dann mit Schritt 5 fort.

Systemstart

Befehlsablauf

Schritt 5: Nach einigen Sekunden erscheint dann die MS-DOS-Aufforderung, das Tagesdatum einzugeben:

```
Current date is Tue 1-01-1980
Enter new date:_
```

Schritt 6: Am Ende der Zeile: Enter new date: befindet sich jetzt der Cursor und zeigt Ihnen an, daß Sie hier das Datum eingeben sollen. Das Datum wird in der *amerikanischen Form* eingegen, in der Reihenfolge:

mm-tt-jj (Monat-Tag-Jahr)

Der **Monat** wird mit einer Zahl von 1 bis 12 einschließlich eingegeben.

Der **Tag** wird mit einer Zahl von 1 bis 31 einschließlich eingegeben.

Für den Zeitraum von 1980 bis 1999 wird das **Jahr** mit einer zweistelligen Zahl eingegeben: 80 bis 99.

Für den Zeitraum von 1980 bis 2099 wird das **Jahr** mit einer vierstelligen Zahl eingegeben: 1980 bis 2099.

Die Ziffern werden jeweils durch einen Bindestrich getrennt. Nach Eingabe des Datums betätigen Sie die Taste **RETURN**.

Ein Beispiel für die Eingabe des 1. August 1984:

```
08-01-84
```

Nach der Datumseingabe werden Sie nach der Zeit gefragt. Es erscheint folgendes:

```
Current time is 0:02:35.87
Enter new time:_
```

Schritt 7: Am Ende der Zeile: Enter new time: befindet sich jetzt der Cursor und zeigt Ihnen an, daß Sie hier die Uhrzeit eingeben sollen. Die Zeit wird eingegen in der Reihenfolge:

hh:mm:ss

Die **Stunden** werden mit einer Zahl von 0 bis 23 einschließlich eingegeben.

Die **Minuten** werden mit einer Zahl von 0 bis 59 einschließlich eingegeben.

Die **Sekunden** werden mit einer Zahl von 0 bis 59 einschließlich eingegeben.

Werden keine Minuten oder Sekunden eingegeben, wird als Wert jeweils 00 angenommen. Die Zahlen werden durch einen Doppelpunkt getrennt. Nach Eingabe der Zeit betätigen Sie die Taste **RETURN**.

Ein Beispiel für die Eingabe von halb fünf Uhr nachmittags:

16:30:00 (16:30)

Schritt 8: Nach der Eingabe der Zeit erscheint ein Copyrightvermerk und darunter das Systemanforderungszeichen.

```
MS-DOS Version 3.10
Copyright (C) 1981,1982,1983,1984,1985
Microsoft Corp.
A>
```

Damit ist MS-DOS bereit, Ihre Befehle entgegenzunehmen.

Befindet sich auf Ihrer Startdiskette die Datei AUTOEXEC.BAT, so werden die dort gespeicherten Befehle abgearbeitet, und es wird weder das Datum noch die Zeit abgefragt.

Befehlsablauf

Falls Sie keine neue Eingabe für Datum und Zeit vornehmen wollen, übergehen Sie die Abfragen jeweils durch einfaches Betätigen der Taste **RETURN**. Es werden dann die vorhandenen Werte übernommen. Es ist später jederzeit möglich, diese Werte zu korrigieren. Näheres dazu im Kapitel 11.

Ab der Version 3 ist es möglich, einen landesspezifischen Datensatz zu laden, der eine andere Eingabe gestattet. Geben Sie dafür den Befehl COUNTRY = 49 in der Datei CONFIG.SYS ein. Damit können Sie das Datum in der gewohnten Weise eingeben. Das Programm SELECT erstellt diese Datei automatisch zusammen mit dem Anpassungsprogramm für eine Tastatur nach deutscher DIN-Norm. Weitere Einzelheiten dazu finden Sie in den Kapiteln 3, 9 und 11.

Kapitel 3

Sicherungskopien

Übersicht

In diesem Kapitel wird Ihnen erklärt, wie Sie Kopien von bereits beschriebenen Disketten anfertigen. Dieses gilt sowohl für die mitgelieferten Systemdisketten, als auch für jede andere beschriebene Diskette. Am Ende des Kapitels sollten Sie wenigstens alle Systemdisketten einmal (besser wäre zweimal) kopiert haben. Verwahren Sie dann Ihre *Originalsystemdisketten,* als Masterdisketten, an einem sicheren Ort. Falls Ihre Kopien nicht mehr in Ordnung sind, können Sie davon jederzeit neue Kopien anfertigen.

Disketten mit von Ihnen erfaßten Daten werden genauso kopiert. Sie sollten immer alle Daten doppelt aufbewahren. Dies ist Ihre beste Versicherung gegen unvorhergesehenen Datenverlust. Täglich veränderte Dateien können Sie mit dem Programm BACKUP jederzeit problemlos sichern. Weitere Einzelheiten dazu finden Sie im Kapitel 11.

Außer Ihrem Zenith PC benötigen Sie jetzt die mitgelieferten Originalsystemdisketten, weitere zu kopierende Disketten und genau so viele nicht beschriebene Disketten.

Kopieren der MS-DOS-Disketten

Mit diesem Handbuch haben Sie ebenfalls zwei Disketten bekommen, auf denen das *Betriebssystem* MS-DOS der Version 3.1 enthalten ist. Bevor Sie Ihren Zenith PC benutzen, sollten Sie diese Disketten kopieren. Das neue Dienstprogramm SELECT enthält alle notwendigen Befehle dafür und führt sie automatisch aus. Zusätzlich erstellt SELECT zwei neue Dateien, die das Betriebssystem für den Systemstart benötigt. Die neue Datei CONFIG.SYS enthält den Befehl COUNTRY, der Datum und Uhrzeit auf landesspezifisches Format ändert. Die Datei AUTOEXEC.BAT enthält einen Befehl für den Aufruf eines Tastaturanpassungsprogrammes.

Zum Kopieren der MS-DOS-Disketten benötigen Sie:

- Zenith PC
- zwei original MS-DOS-Systemdisketten
- zwei neue unbespielte Disketten

Ist Ihr Zenith PC mit *Diskettenlaufwerken* unterschiedlicher Typen ausgerüstet, beachten Sie zusätzlich folgende Beschränkungen:

- Ist Ihr Gerät mit einem High-Density-Laufwerk und einem herkömmlichen Laufwerk ausgerüstet, können Sie die Disketten nicht mehr beliebig tauschen. Disketten des High-Density-Laufwerkes können nicht mehr vom herkömmlichen Laufwerk gelesen werden. Haben Sie in diesem eine Diskette formatiert und bespielen diese Diskette in einem High-Density-Laufwerk, kann die Diskette nicht mehr vom herkömmlichen Laufwerk gelesen werden.

Beschriften Sie die Disketten mit den Sicherungskopien, beachten Sie die Vorsichtsmaßnahmen, die im Kapitel 1, Seite 7 beschrieben werden.

SELECT

Ihre MS-DOS-*Systemdisketten* können Sie entweder mit dem Befehl DISKCOPY oder dem Befehl SELECT kopieren. Der Befehl SELECT erstellt gleichzeitig zwei Dateien auf der neuen Diskette, die das Betriebssystem während des Bootens abarbeitet. Damit werden Datum und Zeit im gewohnten Format dargestellt, und die Tastatur entspricht der deutschen DIN-Norm. Legen Sie in das Laufwerk A: die original Systemdiskette 1 (Schreibschutz nicht vergessen!) und in das Laufwerk B: eine neue noch nicht benutzte Diskette. Rufen Sie den Befehl SELECT mit folgender Eingabezeile auf:

```
SELECT yyy xxxx
```

dabei bedeutet:

yyy Landeskennziffer. Damit wird das Format von Datum und Uhrzeit bestimmt.

xxxx Landesname. Damit wird das Tastaturanpassungsprogramm bestimmt.

Wählen Sie die Werte für die Landeskennziffer und den Namen aus der Tabelle 3.1.

Tabelle 3.1 Zusatzeingaben für SELECT

LAND	Kennziffer	Tastatur 7-Bit	Tastatur 8-Bit
Australien	061	AUS	US
Belgien	032	AFR	FR
Dänemark	045	ADA	DA
Deutschland	049	AGR	GR
Finnland	358	ASW	SW
Frankreich	033	AFR	FR
Großbritanien, UK	044	AUK	UK
Großbritanien, US	044	AUS	US

Sicherungskopien

Kopieren der MS-DOS-Disketten

Tabelle 3.1 Zusatzeingaben für SELECT

LAND	Kennziffer	Tastatur 7-Bit	8-Bit
Israel	972	---	---
Italien	039	AIT	IT
Kanada, EL	001	---	CANE
Kanada, SE	001	---	CANS
Niederlande	031	AUS	US
Norwegen	047	ANO	NO
Schweiz, DT	041	---	CHG
Schweiz, FR	041	---	CHF
Spanien	034	ASP	SP
USA	001	AUS	US

Für ein deutsches Betriebssystem rufen Sie das Programm SELECT mit folgendem Befehl auf:

```
SELECT 049 GR
```

Das *Tastaturprogramm* hat den kompletten Namen KEYBGR.COM und sollte immer benutzt werden. Die 7-Bit-Fassung mit dem Namen KEYBAGR.COM benötigen Sie für einen 7-Bit-Drucker, der deutsche Sonderzeichen an Stelle der amerikanischen Sonderzeichen darstellen kann. Benutzen Sie das Programm KEYBAGR.COM, werden nicht mehr von allen Programmen deutsche Sonderzeichen auf dem Bildschirm dargestellt. Dies gilt immer dann, wenn diese Programme Zeichen direkt darstellen und das Betriebssystem nicht benutzen.

Der Ablauf des Programmes SELECT entspricht den Erklärungen des nächsten Abschnittes: Kopieren von Disketten.

Kopieren von Disketten

Mit dem Befehl DISKCOPY werden Disketten komplett kopiert. Da es ein externer Befehl ist, ist auf Ihrer Kopie der ersten Systemdiskette ein eigenes Programm mit dem Namen DISKCOPY.COM vorhanden. Wollen Sie andere Disketten als die Systemdiskette 1 kopieren, so müssen Sie dieses Dienstprogramm erst in den Arbeitsspeicher laden (durch Eingabe von: DISKCOPY **RETURN**), bevor Sie Disketten kopieren können. Danach können Sie diese Diskette wieder entfernen.

Sie sollten unbedingt vorher die Ursprungsdiskette (Diskette mit den zu kopierenden Daten) gegen unbeabsichtigtes Schreiben schützen. Überkleben Sie dafür die Kerbe an der linken Seite der Diskette mit einem kleinen Streifen.

Sie sollten als Zieldiskette (Diskette mit den kopierten Daten) immer eine neue Diskette nehmen oder eine Diskette, auf der keine Dateien mehr vorhanden sind. Diese Vorsichtsmaßnahmen sind notwendig, da das Dienstprogramm DISKCOPY die Zieldiskette immer neu formatiert.

Der Befehlsablauf für Geräte mit nur einem Diskettenlaufwerk wird ab Seite 2.4 erklärt. Dies gilt auch für Geräte mit einem Diskettenlaufwerk und einem Winchesterlaufwerk, denn der Befehl ist für Winchesterlaufwerke nicht anwendbar.

Kopieren von Disketten

Befehlsablauf für Geräte mit *ZWEI* Diskettenlaufwerken

Schritt 1: Legen Sie die Diskette mit dem Dienstprogramm DISK-COPY.COM in das obere oder linke Diskettenlaufwerk (A:) und geben Sie ein:

```
DISKCOPY/V
```

Danach betätigen Sie die Taste **RETURN**. Durch das nachgestellte „/V" wird der Kopiervorgang vom Betriebssystem laufend überprüft. Es werden die kopierten Daten mit den Ursprungsdaten verglichen und gegebenenfalls wird eine Fehlermeldung ausgegeben (s. a. Kapitel 11). Es erscheint jetzt auf dem Bildschirm folgende Meldung:

```
              DISKCOPY Version 3.10
Copyright (C) 1985 Zenith Data Systems Corporation

Source drive name?  (A-B) :
```

Schritt 2: Auf die Frage nach dem Ursprungslaufwerk (Source drive name?) geben Sie jetzt ein **A** ein. Es erscheint dann die Frage:

```
Destination drive name?   (A-B) :
```

Schritt 3: Auf die Frage nach dem Ziellaufwerk (Destination drive name?) geben Sie jetzt ein **B** ein. Es erscheint jetzt die folgende Meldung:

```
Place the source Disk in A and
the destination Disk in B.
Press RETURN when ready.
```

Sie werden jetzt aufgefordert, die Ursprungsdiskette in das Laufwerk A: (oberes oder linkes Laufwerk) einzulegen, die Zieldiskette in das Laufwerk B: (unteres oder rechtes Laufwerk) einzulegen. Danach betätigen Sie die Taste **RETURN**.

Kopieren von Disketten

Schritt 4: Die beiden Kontrollampen der Laufwerke leuchten jetzt auf, und es erscheint die folgende Meldung:

Formatting destination...

Damit wird angezeigt, daß die Zieldiskette neu formatiert wird. Sollten noch Dateien auf der Diskette vorhanden sein, werden sie jetzt gelöscht. Der bereits erwähnte Schreibschutz auf der Ursprungsdiskette dient als Sicherheit dafür, daß diese Diskette nicht versehentlich als Zieldiskette neu formatiert wird.

Schritt 5: Nach einigen Sekunden erscheint die Meldung:

Copying...

Damit wird angezeigt, daß die Dateien kopiert werden.

Schritt 6: Nach einigen Sekunden erscheint die Meldung:

Verifying...

Damit wird angezeigt, daß die kopierten Dateien mit den Ursprungsdateien verglichen werden. Sollte eine Datei nicht richtig kopiert worden sein, erscheint eine Fehlermeldung (s. a. Kapitel 11).

Schritt 7: Nach einigen Sekunden erscheint die Meldung:

Do you wish to copy another disk (Y/N)? <N>

Sie werden gefragt, ob Sie eine weitere Diskette kopieren wollen. Geben Sie ein **Y** für Ja ein, beginnt der Vorgang erneut. Geben Sie ein **N**(ein) ein, verlassen Sie das Programm, und es erscheint das Systemanforderungszeichen.

Befehlsablauf für Geräte mit *EINEM* Diskettenlaufwerk

Schritt 1: Legen Sie die Diskette mit dem Dienstprogramm DISK-COPY.COM in das obere Diskettenlaufwerk (A:) und geben Sie ein:

DISKCOPY/V

Danach betätigen Sie die Taste **RETURN**. Durch das nachgestellte „/V“ wird der Kopiervorgang vom Betriebssystem laufend überprüft. Dabei werden die kopierten Daten mit den Ursprungsdaten verglichen und gegebenenfalls eine Fehlermeldung ausgegeben (s. a. Kapitel 11). Es erscheint jetzt auf dem Bildschirm folgende Meldung:

```
DISKCOPY Version 3.10
Copyright (C) 1985 Zenith Data Systems Corporation

Source drive name?   (A-B) :
```

Schritt 2: Auf die Frage nach dem Ursprungslaufwerk (Source drive name?) geben Sie jetzt ein **A** ein. Es erscheint dann die Frage:

```
Destination drive name?   (A-B) :
```

Schritt 3: Auf die Frage nach dem Ziellaufwerk (Destination drive name?) geben Sie jetzt ein **B** ein. Jetzt wird das physikalische Laufwerk A auch als imaginäres Laufwerk B behandelt. Es wird dann nicht Laufwerk A: oder B: geschrieben, sondern immer Diskette A: oder B:. Sie werden dann, falls erforderlich, jedesmal aufgefordert, die Disketten zu vertauschen (Schreibschutz Diskette A !!). Es erscheint jetzt die folgende Meldung:

```
Place source the Disk in A and
the destination Disk in B.
Press RETURN when ready.
```

Kopieren von Disketten

Schritt 4: Jetzt werden Sie aufgefordert, die Ursprungsdiskette in das Laufwerk A: einzulegen. Entfernen Sie deshalb jetzt die Diskette mit dem Dienstprogramm. Danach betätigen Sie die Taste **RETURN**.

Schritt 5: Es erscheint die folgende Meldung:

```
Formatting destination...Place disk B in drive A
Press any key when ready
```

Damit wird angezeigt, daß die Zieldiskette neu formatiert wird. Sollten noch Dateien auf der Diskette vorhanden sein, werden sie jetzt gelöscht. Der bereits erwähnte Schreibschutz auf der Ursprungsdiskette dient als Sicherheit dafür, daß diese Diskette nicht versehentlich als Zieldiskette neu formatiert wird.

Tauschen Sie jetzt die Disketten aus. Danach betätigen Sie eine beliebige Taste.

Schritt 6: Nach einigen Sekunden erscheint die Meldung:

```
Copying...Place disk A in drive A
Press any key when ready
```

Damit wird angezeigt, daß die Dateien jetzt kopiert werden können. Tauschen Sie die Disketten aus, und betätigen Sie irgendeine Taste.

Je nach Anzahl der zu kopierenden Dateien, werden Sie jetzt wiederholt aufgefordert, die Disketten zu tauschen.

Kopieren von Disketten

Schritt 7: Ist der Kopiervorgang beendet, erscheint die Meldung:

```
Verifying...Place disk A in drive A
Press any key when ready
```

Damit wird angezeigt, daß die kopierten Dateien mit den Ursprungsdateien verglichen werden. Sollte eine Datei nicht richtig kopiert worden sein, erscheint eine Fehlermeldung (s. a. Kapitel 11). Tauschen Sie nochmals die Disketten.

Schritt 8: Nach einigen Sekunden erscheint die Meldung:

```
Do you wish to copy another disk (Y/N)? <N>
```

Sie werden gefragt, ob Sie eine weitere Diskette kopieren wollen. Geben Sie ein **Y** für Ja ein, beginnt der Vorgang erneut. Geben Sie ein **N**(ein) ein, verlassen Sie das Programm, und es erscheint das Systemanforderungszeichen.

Kapitel 4

Arbeitskopien

Übersicht

Bei intensivem Einsatz des Betriebssystems zur Unterstützung der Anwenderprogramme ist es von Vorteil, die am meisten benutzten Dienstprogramme auf die jeweilige Arbeitsdiskette zu kopieren. Die Programme müssen vor jedem Aufruf in den Arbeitsspeicher geladen werden. Diese Programme werden anschließend von folgenden Programmen wieder überschrieben und gelöscht. Damit sie aufgerufen werden können, müssen sie auf der Arbeitsdiskette gespeichert sein. Ist dies nicht der Fall, erscheint die Fehlermeldung:

Bad command or filename	Befehl oder Datei falsch

Ist Ihr Zenith PC mit einem Winchesterlaufwerk ausgerüstet, können Sie auch sämtliche MS-DOS-Dienstprogramme in einem eigenen Inhaltsverzeichnis abspeichern (s. a. Kapitel 7).

Im ersten Teil dieses Kapitels wird ausführlich erklärt, wie Sie eine Arbeitsdiskette erstellen. Im zweiten Teil wird ausführlich erklärt, wie Sie ein Winchesterlaufwerk aufteilen und anschließend einen Teil als Arbeitsbereich nutzen.

Erstellen einer Arbeitsdiskette

Für das Erstellen einer Arbeitsdiskette benötigen Sie eine Sicherungskopie Ihrer Systemdiskette. Auf dieser müssen außer den Systemdateien die folgenden Dienstprogramme enthalten sein:

COMMAND.COM, FORMAT.COM und CONFIGUR.COM

Desweiteren benötigen Sie eine Diskette mit Kopien Ihrer Anwenderprogramme und mehrere leere Disketten.

In diesem Abschnitt werden die folgenden Schritte erklärt:

- FORMAT
- COPY
- CONFIGUR

Benutzen Sie ein High-Density-Laufwerk zur Herstellung Ihrer Arbeitsdiskette, beachten Sie, daß Sie eine Diskette mit 96-tpi verwenden. Soll diese Diskette von einem normalen Laufwerk gelesen werden, ist es erforderlich, das Programm FORMAT mit dem Zusatz /4 aufzurufen.

FORMAT

Das Dienstprogramm FORMAT bereitet Ihre Diskette zur Speicherung der Daten vor (s. a. Kapitel 11). Bevor Sie es aufrufen, bereiten Sie die zu kopierenden Disketten vor. Sie sollten auf jeder Diskette ein Etikett anbringen, auf dem Sie vermerken, wann Sie die Diskette mit welchem Betriebssystem formatiert haben. Verwenden Sie nicht Dienstprogramme der verschiedenen Versionen gleichzeitig.

Schritt 1: Rufen Sie das Programm folgendermaßen auf:

FORMAT B:/S **RETURN**

Hinter dem Programm- oder Befehlsnamen wird das Laufwerk mit der zu formatierenden Diskette eingegeben. Geben Sie ein „ **/S** " zusätzlich ein, wird nach dem Formatieren das komplette Betriebssystem auf die neue Diskette kopiert. Arbeiten Sie mit einem High-Density-Laufwerk, können Sie mit dem weiteren Zusatz /4, die Diskette im bisherigen Format mit 360 Kb Speichervermögen vorbereiten. Geben Sie dann ein:

FORMAT B:/S/4 **RETURN**

Auf dem Bildschirm erscheint jetzt folgende Meldung:

```
          FORMAT Version 3.01
Copyright (C) 1985, Zenith Data Systems Corporation

Insert new disk in drive B
and press RETURN when ready.
```

Erstellen einer Arbeitsdiskette

Schritt 2: Sie werden aufgefordert, die zu formatierende Diskette in das Laufwerk B: (rechtes Laufwerk) einzulegen.

Ist Ihr Zenith PC mit nur einem Diskettenlaufwerk ausgerüstet, so müssen Sie die Diskette mit dem Dienstprogramm gegen die zu formatierende tauschen. Es erscheint der folgende Text:

```
Place disk B in drive A:
Press any key when ready
```

Legen Sie eine neue Diskette in das Laufwerk, und betätigen Sie danach die Taste **RETURN**.

Schritt 3: Während des Formatierens leuchtet die Kontrollampe des Diskettenlaufwerkes. Zum Schluß werden die Systemdateien auf die neue Diskette kopiert (s. a. Kapitel 11). Es erscheint die Meldung:

```
System transferred

Enter desired volume label (11 characters, RETURN for none)?
```

Sie können der Diskette jetzt einen Namen geben. Dieser Name (maximal elf Zeichen) darf nur aus den für Dateinamen gültigen Zeichen bestehen. Er wird bei jeder Auflistung des Inhaltsverzeichnisses mit angezeigt. Mit dem Programm LABEL können Sie diesen jederzeit ändern oder löschen. Betätigen Sie nur die Taste **RETURN**, wird kein Name gespeichert.

Schritt 4: Es erscheint jetzt die Meldung:

```
362496 bytes total disk space
 41984 bytes used by system
320512 bytes available on disk

Do you wish to format another disk (Y/N)?
```

Mit dieser Statusmeldung zeigt das System an, daß der Kopiervorgang beendet ist. In der obersten Zeile wird die Gesamtkapazität der Diskette genannt. Da die Systemdateien mitkopiert wurden, wird in der zweiten Zeile angezeigt, wieviel Platz diese Dateien belegt haben. Der restliche Speicherplatz der Diskette kann von Ihren Daten beschrieben werden. Diese Zahlen können je nach benutztem Zusatz beim Aufruf des Programmes FORMAT verschieden sein. Sollte die Diskette schadhaft sein, würden ebenfalls die nicht zu beschreibenden Sektoren (bad sektors) angezeigt.

Wollen Sie weitere Disketten formatieren, geben Sie ein **Y** für ja ein, andernfalls wird der Vorgang mit **N**(ein) beendet, und es erscheint wieder das Systemanforderungszeichen des Betriebssystemes.

COPY

Mit dem internen Befehl COPY können Sie jetzt auf die gerade formatierte Diskette Informationen (Programme oder Dateien) umkopieren. Kopieren Sie Anwenderprogramme und die am häufigsten benötigten Dienstprogramme auf diese Diskette, so enthält die Diskette alle notwendigen Programme, die Sie zum Arbeiten brauchen.

Schritt 1: Legen Sie die Diskette mit den zu kopierenden Programmen in das Laufwerk A (oberes Laufwerk). Legen Sie die gerade formatierte Diskette in das Laufwerk B (unteres Laufwerk). Geben Sie den folgenden Befehl ein:

```
COPY DATEINAME.ZUS B:/V
```

Arbeitskopien

Erstellen einer Arbeitskopie

Damit weisen Sie das System an, die genau bezeichnete Datei vom Arbeitslaufwerk (A) auf das Laufwerk B zu kopieren. Der Zusatz „/V“ gibt an, daß die kopierten Daten mit den Ursprungsdaten verglichen werden. Eventuelle Fehler werden angezeigt (s. a. Kapitel 11).

Schritt 2: Es leuchten jetzt kurzzeitig die Kontrollampen der beiden Laufwerke auf. Dies zeigt an, daß die Daten von einem Laufwerk zum anderen kopiert werden. Ist der Vorgang abgeschlossen, erscheint folgende Meldung:

1 File(s) copied	1 Datei kopiert
A>_	A>_

Damit ist der Kopiervorgang beendet. Die einzelnen Schritte sind für jede Datei in der gleichen Weise zu wiederholen.

Mit dem Befehl COPY können Sie jede Datei einzeln kopieren oder auch mehrere Dateien zusammenfassen und automatisch kopieren lassen. Geben Sie z. B. folgenden Befehl ein:

```
COPY A:*.* B:/V
```

Jetzt werden alle Dateien der Diskette des Laufwerkes A: auf die Diskette des Laufwerkes B: kopiert. Weitere Einzelheiten dazu finden Sie im Kapitel 11.

Ist Ihr Zenith PC nur mit einem Diskettenlaufwerk ausgestattet, werden Sie wiederholt aufgefordert, beide Disketten zu vertauschen. Nähere Erläuterungen dazu finden Sie im Kapitel 5.

CONFIGUR

Der externe Befehl CONFIGUR ruft ein Dienstprogramm auf, das die seriellen Ausgänge (COM1: und COM2:) Ihres Zenith PC an die anzuschließenden Geräte (z. B. Drucker oder Modem) anpaßt. Mit diesem Programm können Sie das *Datenübertragungsprotokoll* eines seriellen Ausganges festlegen. Das Protokoll bestimmt die Einzelheiten der Datenübertragung zwischen Computer und Peripheriegerät. Ausführlich wird das Programm im Kapitel 11 beschrieben. Hier ist eine Kurzform wiedergegeben, die es Ihnen ermöglicht, sofort einen seriellen Drucker in Betrieb zu nehmen.

Schließen Sie einen Paralleldrucker an Ihr Gerät an, brauchen Sie dieses Dienstprogramm nicht aufzurufen; ebensowenig wenn der serielle Drucker bereits einwandfrei arbeitet.

Das Programm CONFIGUR.COM befindet sich auf der Kopie der Systemdiskette eins. Es ist menügesteuert. Die von Ihnen zu machenden Eingaben können Sie jedesmal aus den angezeigten Möglichkeiten auswählen. Andere Eingaben sind nicht möglich. Es wird erst der Parallelausgang bestimmt, dann wird der Ausgang umgeleitet zum Seriellausgang.

Schritt 1: Rufen Sie das Programm folgendermaßen auf:

CONFIGUR **RETURN**

Es erscheint dann folgende Meldung:

```
CONFIGUR Version 3.01
Copyright (C) 1985 Zenith Data Systems Corporation

Use one of the following options to configure a device

A: Configure LPT device
B: Configure COM device
C: Change automatic partition assignment flag

D: Exit with no changes

Enter selection (A-D):
```

Erstellen einer Arbeitsdiskette

Schritt 2: Geben Sie ein **A** ein, um den parallelen Ausgang zu definieren. Es erscheint folgendes Menü:

A: Map parallel output to serial output
B: Configure parallel device

C: Exit

Schritt 3: Geben Sie ein **A** ein für die Datenumleitung. Es erscheint folgendes Menü:

A: LPT1
B: LPT2
C: LPT3

D: Exit

Schritt 4: Geben Sie ein **A** ein für den ersten Ausgang. Es erscheint folgendes Menü:

A: No mapping
B: COM1
C: COM2

Schritt 5: Geben Sie ein **B** ein für den ersten Ausgang. Es erscheint folgendes Menü:

A: Map parallel output to serial output
B: Configure parallel device

C: Exit

Schritt 6: Geben Sie ein **C** ein, um das Programm zu verlassen. Es erscheint wieder das Hauptmenü, erweitert um einige Möglichkeiten:

A: Configure LPT device
B: Configure COM device
C: Change automatic partition assignment flag

Use one of the following to modify an existing system

D: Exit programm
E: Make changes to disk
F: Make changes to memory
G: Make changes to both disk and memory

Erstellen einer Arbeitsdiskette

Schritt 7: Geben Sie jetzt ein **B** ein für den seriellen Ausgang. Es erscheint folgendes Menü:

A: COM1
B: COM2

C: Exit

Schritt 8: Geben Sie jetzt ein **A** ein für den ersten (oberen) Seriellausgang. Es erscheint eine Auswahl von bereits gespeicherten Geräten:

A. Compatibility mode (2400 baud, DTR and RTS pos.)
B. MX-80 (4800 baud, DTR pos, (pin 20)
C. H/Z-25 (4800 baud, RTS pos, (pin 4)
D. H-14/WH-24 (4800 baud, RTS Neg, (pin4))
E. Diablo 630/1640 (1200 baud , ETX/ACK)
F. WH-23/WH-33/WH-43 modem (300 baud, No handshake)
G. WH-12 Votrax Type-N-Talk (4800 baud, RTS Pos. (pin45))
H. User Defined
I. Exit with no changes

Schritt 9: Wählen Sie hier Ihr Gerät aus oder falls nicht vorhanden, geben Sie ein **H** ein. Weitere Erläuterungen zur Möglichkeit H finden Sie dann im Kapitel 11.

Schritt 10: Nach Auswahl des Gerätes oder Festlegung des eigenen Protokolles erscheint wieder das Hauptmenü:

A: Configure LPT device
B: Configure COM device
C: Change automatic partition assignment flag

Use one of the following to modify an existing system

D: Exit programm
E: Make changes to disk
F: Make changes to memory
G: Make changes to both disk and memory

Arbeitskopien

Erstellen einer Arbeitsdiskette

Schritt 11: Geben Sie jetzt ein **G** ein, damit die Änderungen vom System erkannt werden; ebenso werden sie dann auf der Diskette gespeichert. Es erscheint folgende Frage:

```
Enter drive name with system to be modify (A-B):
```

Schritt 12: Geben Sie hier den Namen des Laufwerks ein, in dem sich eine Systemdiskette befindet, die modifiziert werden soll. Sie können vorher noch die Diskette auswechseln. Danach erscheint wieder das Hauptmenü.

Schritt 13: Sie können jetzt weitere Disketten modifizieren, oder das Programm mit Eingabe von **D** verlassen.

Es erscheint jetzt wieder das Systemanforderungszeichen. Haben Sie einen Drucker am ersten seriellen Ausgang angeschlossen (s. a. Gerätehandbuch), können Sie durch Betätigen der Tasten **SHIFT-PTRSC** den gesamten Bildschirminhalt ausdrucken lassen.

Arbeitet der Drucker nicht, ist es möglich, daß Ihr Gerät keine weiteren Befehle mehr entgegennimmt. Dann ist nichts weiter passiert, als daß der Drucker nicht richtig angepaßt wurde. Starten Sie Ihr System neu, und wiederholen Sie den Vorgang. Eventuell wird es notwendig sein, im Kapitel 11 ausführlichere Informationen über das Programm CONFIGUR nachzulesen.

Aufteilen eines Winchesterlaufwerkes

Für das Aufteilen eines Winchesterlaufwerkes benötigen Sie eine Sicherungskopie Ihrer Systemdiskette. Auf dieser müssen außer dem System Version Nr. 3.10 folgende Dienstprogramme enthalten sein:

COMMAND.COM, FORMAT.COM und CONFIGUR.COM

Desweiteren benötigen Sie eine Diskette mit Kopien Ihrer Anwenderprogramme und mehrere leere Disketten.

In diesem Abschnitt werden folgende Schritte erklärt:

- FORMAT
- COPY
- CONFIGUR

Sie können auf das Winchesterlaufwerk nur mit dem Betriebssystem MS-DOS ab der Version Nr. 2.0 zugreifen.

Aufteilen eines Winchesterlaufwerkes

FORMAT

Das Dienstprogramm FORMAT bereitet Ihr Winchesterlaufwerk zur Speicherung der Daten vor (s. a. Kapitel 11). Es kopiert ebenfalls automatisch die Systemdateien auf das Laufwerk.

Schritt 1: Rufen Sie das Programm folgendermaßen auf:

FORMAT C:/S **RETURN**

Hinter dem Programm- oder Befehlsnamen wird zuerst die Laufwerkbezeichnung eingegeben. Geben Sie ein „ **/S** “ zusätzlich ein, wird nach dem Formatieren das komplette Betriebssystem auf das Laufwerk kopiert. Auf dem Bildschirm erscheint jetzt folgende Meldung:

```
FORMAT Version 3.01
Copyright (C) 1985, Zenith Data Systems Corporation

Will FORMAT partition assigned drive C
Press RETURN when ready.
```

Schritt 2: Betätigen Sie danach die Taste **RETURN**.

Schritt 3: Während des Formatierens leuchtet die Kontrollampe des Winchesterlaufwerkes auf. Zum Schluß werden die Systemdateien auf die neue Diskette kopiert (s. a. Kapitel 11). Es erscheint die Meldung:

```
System transferred

Enter desired volume label (11 characters, RETURN for none)?
```

Aufteilen eines Winchesterlaufwerkes

Sie können dem Laufwerkteil jetzt einen Namen geben. Dieser Name (maximal elf Zeichen) darf nur aus den für Dateinamen gültigen Zeichen bestehen. Er wird bei jeder Auflistung des Inhaltsverzeichnisses mit angezeigt. Mit dem Programm LABEL können Sie diesen jederzeit ändern oder löschen. Betätigen Sie nur die Taste **RETURN**, wird kein Name gespeichert.

Schritt 4: Es erscheint jetzt die Meldung:

```
2662400 bytes total disk space
  41984 bytes used by system
2620416 bytes available on disk
```

Mit dieser Statusmeldung zeigt das System an, daß der Kopiervorgang beendet ist. In der obersten Zeile wird die Gesamtkapazität des Laufwerkes genannt. Da die Systemdateien mitkopiert wurden, wird in der zweiten Zeile angezeigt, wieviel Platz diese Dateien belegt haben. Der restliche Speicherplatz des Laufwerkes kann von Ihren Daten beschrieben werden. Diese Zahlen können je nach Größe des Laufwerkes und benutztem Zusatz beim Aufruf des Programmes FORMAT verschieden sein.

Es erscheint jetzt wieder das Systemanforderungszeichen. Damit ist die Formatierung abgeschlossen, und Sie können Daten auf Ihrer Winchester speichern.

COPY

Mit dem internen Befehl COPY können Sie jetzt auf den gerade formatierten Teil Ihrer Winchester Informationen (Programme oder Dateien) umkopieren. Kopieren Sie Anwenderprogramme und die am häufigsten benötigten Dienstprogramme auf diesen Teil, so enthält er alle notwendigen Programme, die Sie zum Arbeiten brauchen.

Schritt 1: Legen Sie die Diskette mit den zu kopierenden Programmen in das Laufwerk A: (oberes Laufwerk). Geben Sie den folgenden Befehl ein:

```
COPY DATEINAME.ZUS C:/V
```

Damit weisen Sie das System an, die genau bezeichnete Datei vom Arbeitslaufwerk (A) auf das Laufwerk C: zu kopieren. Der Zusatz „ /V “ gibt an, daß die kopierten Daten mit den Ursprungsdaten verglichen werden. Eventuelle Fehler werden angezeigt (s. a. Kapitel 11).

Schritt 2: Es leuchten jetzt kurzzeitig die Kontrollampen der beiden Laufwerke. Dies zeigt an, daß die Daten von einem Laufwerk zum anderen kopiert werden. Ist der Vorgang abgeschlossen, erscheint folgende Meldung:

```
1 File(s) copied          1 Datei kopiert
```

Damit ist der Kopiervorgang beendet. Die einzelnen Schritte sind für jede Datei in der gleichen Weise zu wiederholen.

Mit dem Befehl COPY können Sie jede Datei einzeln kopieren oder auch mehrere Dateien zusammenfassen und automatisch kopieren lassen. Geben Sie z. B. folgenden Befehl ein:

```
COPY A:*.* B:/V
```

Jetzt werden alle Dateien der Diskette des Laufwerkes A: auf die Diskette des Laufwerkes B: kopiert. Weitere Einzelheiten dazu finden Sie im Kapitel 11.

CONFIGUR

Der externe Befehl CONFIGUR ruft ein Dienstprogramm auf, das die seriellen Ausgänge (COM1: und COM2:) Ihres Zenith PC an die anzuschließenden Geräte (z. B. Drucker oder Modem) anpaßt. Mit diesem Programm können Sie das *Datenübertragungsprotokoll* eines seriellen Ausganges festlegen. Das Protokoll bestimmt die Einzelheiten der Datenübertragung zwischen Computer und Peripheriegerät. Ausführlich wird das Programm im Kapitel 11 beschrieben. Hier ist eine Kurzform wiedergegeben, die es Ihnen ermöglicht, sofort einen seriellen Drucker in Betrieb zu nehmen.

Schließen Sie einen Paralleldrucker an Ihr Gerät an, brauchen Sie dieses Dienstprogramm nicht aufzurufen; ebensowenig wenn der serielle Drucker bereits einwandfrei arbeitet.

Das Programm CONFIGUR.COM befindet sich auf der Kopie der Systemdiskette eins. Es ist menügesteuert. Die von Ihnen zu machenden Eingaben können Sie jedesmal aus den angezeigten Möglichkeiten auswählen. Andere Eingaben sind nicht möglich. Es wird erst der Parallelausgang bestimmt, dann wird dieser Ausgang zum Seriellausgang umgeleitet.

Schritt 1: Rufen Sie das Programm folgendermaßen auf:

CONFIGUR **RETURN**

Es erscheint dann folgende Meldung:

```
                CONFIGUR Version 3.01
Copyright (C) 1985 Zenith Data Systems Corporation

Use one of the following options to configure a device

A: Configure LPT device
B: Configure COM device
C: Change automatic partition assignment flag

D: Exit with no changes

Enter selection (A-D):
```

Arbeitskopien

Aufteilen eines Winchesterlaufwerkes

Schritt 2: Geben Sie ein **A** ein, um den parallelen Ausgang zu definieren. Es erscheint folgendes Menü:

A: Map parallel output to serial output
B: Configure parallel device
C: Exit

Schritt 3: Geben Sie ein **A** ein für die Datenumleitung. Es erscheint folgendes Menü:

A: LPT1
B: LPT2
C: LPT3

D: Exit

Schritt 4: Geben Sie ein **A** ein für den ersten Ausgang. Es erscheint folgendes Menü:

A: No mapping
B: COM1
C: COM2

Schritt 5: Geben Sie ein **B** ein für den ersten Ausgang. Es erscheint folgendes Menü:

A: Map parallel output to serial output
B: Configure parallel device

C: Exit

Aufteilen eines Winchesterlaufwerkes

Schritt 6: Geben Sie ein **C** ein, um das Programm zu verlassen. Es erscheint wieder das Hauptmenü, erweitert um einige Möglichkeiten:

A: Configure LPT device
B: Configure COM device
C: Change automatic partition assignment flag

Use one of the following to modify an existing system

D: Exit programm
E: Make changes to disk
F: Make changes to memory
G: Make changes to both disk and memory

Schritt 7: Geben Sie jetzt ein **B** ein für den seriellen Ausgang. Es erscheint folgendes Menü:

A: COM1
B: COM2

C: Exit

Schritt 8: Geben Sie jetzt ein **A** ein für den ersten (oberen) Seriellausgang. Es erscheint eine Auswahl von bereits gespeicherten Geräten:

A. Compatibility mode (2400 baud, DTR and RTS pos.)
B. MX-80 (4800 baud, DTR pos, (pin 20)
C. H/Z-25 (4800 baud, RTS pos, (pin 4)
D. H-14/WH-24 (4800 baud, RTS Neg, (pin4))
E. Diablo 630/1640 (1200 baud , ETX/ACK)
F. WH-23/WH-33/WH-43 modem (300 baud, No handshake)
G. WH-12Votrax Type-N-Talk (4800 baud, RTS Pos. (pin45))
H. User Defined
I. Exit with no changes

Schritt 9: Wählen Sie hier Ihr Gerät aus oder falls nicht vorhanden, geben Sie ein **H** ein. Weitere Erläuterungen zur Möglichkeit H finden Sie dann im Kapitel 11.

Aufteilen eines Winchesterlaufwerkes

Schritt 10: Nach Auswahl des Gerätes oder Festlegung des eigenen Protokolles erscheint wieder das Hauptmenü:

```
A: Configure LPT device
B: Configure COM device
C: Change automatic partition assignment flag

Use one of the following to modify an existing system

D: Exit programm
E: Make changes to disk
F: Make changes to memory
G: Make changes to both disk and memory
```

Schritt 11: Geben Sie jetzt ein **G** ein, damit die Änderungen vom System erkannt werden; ebenso werden sie dann auf Diskette gespeichert. Es erscheint dann folgende Frage:

```
Enter drive name with system to be modify (A-C):
```

Schritt 12: Geben Sie hier den Namen des Laufwerks ein, in dem sich eine Systemdiskette befindet, die modifiziert werden soll. Sie können vorher noch die Diskette auswechseln. Danach erscheint wieder das Hauptmenü.

Schritt 13: Sie können jetzt weitere Disketten modifizieren oder das Programm mit Eingabe von C verlassen.

Es erscheint jetzt wieder das Systemanforderungszeichen. Haben Sie einen Drucker am ersten seriellen Ausgang angeschlossen (s. a. Gerätehandbuch), können Sie durch Betätigen der Tasten **SHIFT-PTRSC** den gesamten Bildschirminhalt ausdrucken lassen.

Arbeitet der Drucker nicht, ist es möglich, daß Ihr Gerät keine weiteren Befehle mehr entgegennimmt. Dann ist nichts weiter passiert, als daß der Drucker nicht richtig angepaßt wurde. Starten Sie Ihr System neu, und wiederholen Sie den Vorgang. Eventuell wird es notwendig sein, im Kapitel 11 ausführlichere Informationen über das Programm CONFIGUR nachzulesen.

Systemstart vom Winchesterlaufwerk

Haben Sie die einzelnen Schritte der vorherigen Abschnitte durchgeführt, können Sie jetzt Ihren Zenith PC vom Winchesterlaufwerk aus starten. Es ist dann keine Systemdiskette mehr erforderlich.

Schritt 1: Betätigen Sie die Tasten **CTRL, ALT** und **DEL** gleichzeitig. Lassen Sie alle Tasten auch gleichzeitig wieder los. Wurden die Schalter für automatischen Ladevorgang (s. a. Kapitel 6) nicht verändert, startet das System jetzt selbstständig von der Winchester aus. Fahren Sie dann mit Schritt 4 fort.

Schritt 2: Erfolgt der Systemstart nicht automatisch, erscheint jetzt in der linken oberen Ecke Ihres Bildschirmes die Statusmeldung des Monitorprogrammes. Das Anforderungszeichen zur Befehlseingabe ist ein Pfeil: „-->". Die Meldung sieht folgendermaßen aus:

```
MFM-200 Monitor, Version 0.8
Memory Size: 512K bytes
Enter "?" for help
-->
```

In der ersten Zeile steht der Name des Programmes sowie die Versionsnummer. In der zweiten Zeile wird der verfügbare Arbeitsspeicher angezeigt. In der dritten Zeile erscheint als Gedächnisstütze das Fragezeichen, welches Sie eingeben können, um eine Auswahl der Eingabemöglichkeiten des Monitorprogrammes zu erhalten. Nähere Einzelheiten dazu finden Sie im Gerätehandbuch. In der vierten Zeile sehen Sie das Anforderungszeichen für Ihre Eingabe.

Systemstart vom Winchesterlaufwerk

Schritt 3: Geben Sie jetzt ein:

```
BW0:1
```

Sie weisen damit das System an, von der Winchester aus zu booten (**B** = Boot; **W** = Winchester). Die erste angeschlossene Winchester wird intern mit der Nummer **0** bezeichnet, der erste formatierte Teil wurde von Ihnen mit C: bezeichnet. Zwischen der Winchesternummer und der Teilnummer ist die Eingabe eines Doppelpunktes (:) erforderlich (s. a. Kapitel 11).

Schritt 4: Nach einigen Sekunden erscheint dann die MS-DOS-Aufforderung, das Tagesdatum einzugeben:

```
Current date is Tue 1-01-1980
Enter new date:_
```

Schritt 5: Am Ende der Zeile: Enter new date: befindet sich jetzt der Cursor und zeigt Ihnen an, daß Sie hier das Datum eingeben sollen. Das Datum wird in der *amerikanischen Form* eingegen, in der Reihenfolge:

mm-tt-jj (Monat-Tag-Jahr)

Der Monat wird mit einer Zahl von 1 bis 12 einschließlich eingegeben.

Der Tag wird mit einer Zahl von 1 bis 31 einschließlich eingegeben.

Für den Zeitraum von 1980 bis 1999 wird das Jahr mit einer zweistelligen Zahl eingegeben: 80 bis 99.

Für den Zeitraum von 1980 bis 2099 wird das Jahr mit einer vierstelligen Zahl eingegeben: 1980 bis 2099.

Die Ziffern werden jeweils durch einen Bindestrich getrennt. Nach Eingabe des Datums betätigen Sie die Taste **RETURN**.

Ein Beispiel für die Eingabe des 1. August 1984:

08-01-84 (08-01-1984)

Nach der Datumseingabe werden Sie nach der Zeit gefragt. Es erscheint folgendes:

```
Current time is 0:02:35.87
Enter new time:_
```

Schritt 6: Am Ende der Zeile: Enter new time: befindet sich jetzt der Cursor und zeigt Ihnen an, daß Sie hier die Uhrzeit eingeben sollen. Die Zeit wird eingegeben in der Reihenfolge:

hh:mm:ss

Die Stunden werden mit einer Zahl von 0 bis 23 einschließlich eingegeben.

Die Minuten werden mit einer Zahl von 0 bis 59 einschließlich eingegeben.

Die Sekunden werden mit einer Zahl von 0 bis 59 einschließlich eingegeben.

Werden keine Minuten oder Sekunden eingegeben, wird als Wert jeweils 00 angenommen. Die Zahlen werden durch einen Doppelpunkt getrennt. Nach Eingabe der Zeit betätigen Sie die Taste **RETURN**.

Systemstart vom Winchesterlaufwerk

Ein Beispiel für die Eingabe von halb fünf Uhr nachmittags:

16:30:00 (16:30) **RETURN**

Schritt 7: Nach der Eingabe der Zeit erscheint ein Copyrightvermerk und darunter das Systemanforderungszeichen.

```
MS-DOS Version 3.10
Copyright (C) 1981,1982,1983,1984,1985
Microsoft Corp.
A>
```

Damit ist MS-DOS bereit, Ihre Befehle entgegenzunehmen.

Befindet sich auf Ihrer Startdiskette die Datei AUTOEXEC.BAT, so werden die dort gespeicherten Befehle abgearbeitet, und es wird weder das Datum noch die Zeit abgefragt.

Falls Sie keine neue Eingabe für Datum und Zeit vornehmen wollen, übergehen Sie die Abfragen jeweils durch einfaches Betätigen der Taste **RETURN**. Es werden dann die vorhandenen Werte übernommen. Es ist später jederzeit möglich, diese Werte zu korrigieren. Näheres dazu im Kapitel 11.

Ab der Version 3 ist es möglich, einen landesspezifischen Datensatz zu laden, der eine andere Eingabe gestattet. Geben Sie dafür den Befehl COUNTRY = 49 in der Datei CONFIG.SYS ein. Damit können Sie das Datum in der gewohnten Weise eingeben. Das Programm SELECT erstellt diese Datei automatisch zusammen mit dem Anpassungsprogramm für eine Tastatur nach deutscher DIN-Norm. Weitere Einzelheiten dazu finden Sie in den Kapiteln 3, 9 und 11.

Teil II

Grundzüge MS-DOS

Grundzüge MS-DOS

Dieser Teil des Handbuches beschreibt auf einfache Weise die Grundzüge des Betriebssystemes MS-DOS.

Es ist nicht unbedingt erforderlich, diesen Teil des Handbuches zu lesen. Die hier beschriebenen Möglichkeiten helfen Ihnen aber, alle Fähigkeiten des Betriebssystemes voll auszunutzen. Einzige Voraussetzung für Leser, die keine Erfahrung mit Mikrocomputern haben, ist, daß sie den ersten Teil dieses Handbuches sorgfältig gelesen haben.

In den folgenden fünf Kapiteln wird alles Wichtige über MS-DOS erklärt:

- Die Befehlseingabe
- Der Systemstart
- Das hierarchische Disketteninhaltsverzeichnis
- Die Ein-/Ausgabe von Daten
- Die einzelnen Bestandteile von MS-DOS

Kapitel 5

Befehle

Übersicht

Ein Befehl ist eine *Anweisung* an das Betriebssystem, Daten und Programme zu erzeugen, zu verändern, zu analysieren oder zu übertragen.

Die Befehlseingabe von MS-DOS wurde so gestaltet, daß Sie wählen können zwischen einer *direkten* oder *automatischen* Befehlseingabe. Sie können dabei immer mit zwei Diskettenlaufwerken arbeiten, selbst dann, wenn Ihr Gerät mit nur einem ausgerüstet ist.

In diesem Kapitel wird ausführlich erklärt, wie die einzelnen Befehle korrekt einzugeben sind. Ebenso werden alle Fehlermeldungen aufgezeigt und Hinweise zur Fehlerbehandlung gegeben.

Befehlsarten

Befehle sind *Anweisungen* an das Betriebssystem, bestimmte Aufgaben durchzuführen. Diese können sich auf Datenverarbeitung, Datentransport oder Datenanalyse beziehen. Die meisten Befehle werden von Ihnen über die *Tastatur* (Konsole) eingegeben. Sie können aber auch in einer speziellen Datei gespeichert werden und rufen sich dann selber auf.

Das Betriebssystem MS-DOS unterscheidet zwei Arten von Befehlen:

- Interne Befehle
- Externe Befehle

Interne Befehle

Interne Befehle sind Befehle, die durch die *Datei COMMAND.COM* verarbeitet werden. Diese Datei wird beim Systemstart geladen und von keinem Programm überschrieben oder gelöscht. Sie können diese Befehle jederzeit eingeben, wenn Sie sich in keinem Anwenderprogramm befinden, sondern auf Betriebssystemebene sind. Dies wird von der Datei COMMAND.COM durch das *Anforderungszeichen* (Prompt) angezeigt. Dieses Prompt besteht aus einem Buchstaben für das Arbeitslaufwerk, gefolgt von einem Zeichen „größer als" (>). Das Betriebssystem MS-DOS kennt folgende interne Befehle:

BREAK
CHDIR (CD)
CLS
COPY
CTTY
DATE
DIR
ECHO
ERASE (DEL)
EXIT
FOR
GOTO
IF
MKDIR (MD)
PATH
PAUSE
PROMPT
REM
RENAME (REN)
RMDIR (RD)
SET
SHIFT
TIME
TYPE
VER
VERIFY
VOL
D:

In diese Liste wurde zur besseren Übersicht ebenfalls der Befehl zum Wechseln des Arbeitslaufwerkes (D:) mit aufgenommen. Durch Eingabe eines einzelnen Buchstabens (von A bis Z), gefolgt von einem Doppelpunkt (:), können Sie jederzeit das Laufwerk wechseln. Rufen Sie ein ungültiges oder nicht vorhandenes Laufwerk auf, erscheint die Fehlermeldung:

Inalid drive specification	Falsche Laufwerkbezeichnung

Haben Sie Ihr System von der Diskette gestartet, erscheint nach der Eingabe des Datums und der Zeit das Prompt:

A>_

Jetzt können Sie Eingaben vornehmen. Sie schreiben jetzt an die Stelle des Bildschirmes, wo Sie den *Cursor* blinken sehen. Der Cursor wird hier durch das Zeichen (_) dargestellt. Sie ändern das Laufwerk folgendermaßen:

A>_	(Prompt)
A>B: **RETURN**	(Eingabe neues Laufwerk)
B>_	(neues Prompt)

Externe Befehle

Externe Befehle sind eigene kleine *Dienstprogramme,* die vor jedem Aufruf in den Arbeitsspeicher geladen werden müssen. Diese Programme werden von folgenden Programmen dann wieder überschrieben und gelöscht. Damit sie aufgerufen werden können, müssen sie auf der Arbeitsdiskette gespeichert sein. Ist dies nicht der Fall, erscheint die Fehlermeldung:

Bad command or filename	Befehl oder Datei falsch

Bei intensivem Einsatz des Betriebssystems zur Unterstützung der Anwenderprogramme ist es von Vorteil, die am meisten benutzten Dienstprogramme auf die jeweilige Arbeitsdiskette zu kopieren. Ist Ihr Zenith PC mit einem Winchesterlaufwerk ausgerüstet, können Sie auch sämtliche MS-DOS Dienstprogramme in einem eigenen Inhaltsverzeichnis abspeichern (s. a. Kapitel 7).

Befehle

Befehlsarten

Auf Ihren Systemdisketten sind folgende externe Befehle oder Dienstprogramme enthalten:

APPLY	EDLIN	PREP
ASGNPART	EXE2BIN	PRINT
ASSIGN	FC	PSC*Drucker*
ATTRIB	FIND	RDCPM
BACKUP	FORMAT	RECOVER
BOOTF	GRAFTABL	RESTORE
CHKDSK	JOIN	SEARCH
CIPHER	KEYBxxxx	SELECT
COMMAND	LABEL	SHARE
COMP	LIB	SHIP
CONFIGUR	LINK	SORT
DEBUG	MODE	SUBST
DETECT	MORE	SYS
DISKCOMP	NODEBUG	TREE
DISKCOPY	PART	

Diese Dateien sind mit den Namenszusätzen .COM oder .EXE versehen. Außer diesen Zusätzen kennt MS-DOS für externe Befehle noch den Zusatz .BAT. Damit werden Dateien zur *automatischen Befehlsverarbeitung* (Batchdateien) bezeichnet. Diese Datei wird in einem der folgenden Abschnitte erklärt.

Befehlszeile

Sämtliche Befehle werden in einer Zeile, der *Befehlszeile*, direkt nach dem Prompt eingegeben. Dabei gelten für interne Befehle andere Regeln als für externe.

Interne Befehle werden immer ohne vorangestellte Laufwerkbezeichnung eingegeben. Eine eventuelle Änderung des Laufwerkes ist vorher getrennt vorzunehmen. Bestimmte Zusatzfunktionen des internen Befehles werden mit Hilfe eines nachgestellten Buchstabens aufgerufen.

Externen Befehlen kann das Laufwerk vorangestellt werden, auf denen sich das Dienstprogramm befindet. Der Namenszusatz des Programmes (.COM, .EXE oder .BAT) wird nicht mit eingegeben. Befindet sich das Programm nicht im aktuellen Inhaltsverzeichnis, kann der jeweilige Verzeichnisname vorangestellt werden. Bestimmte Zusatzfunktionen des externen Befehles werden mit Hilfe eines nachgestellten Buchstabens oder Buchstabenkombination aufgerufen.

Der interne Befehl DIR zeigt alle externen Befehle des aktuellen Inhaltsverzeichnisses. Eine Übersicht der verfügbaren internen Befehle können Sie damit nicht erhalten.

Aufbau der Befehlszeile

Ihre Eingaben werden vom Betriebssystem in einem besonderen Zwischenspeicher festgehalten und gleichzeitig auf dem Bildschirm angezeigt. Die Anzeige erfolgt dabei immer nach dem Systemprompt in der gleichen Zeile. Daher wird diese Zeile auch Eingabezeile oder Befehlszeile genannt. Eine Befehlszeile besteht meistens aus drei Teilen:

- Funktion
- Variable(n)
- RETURN

Befehlszeile

Auch wenn Sie Eingaben oder Befehle in verschiedenen Formen eingeben dürfen, entsprechen diese Eingaben immer dem oben beschriebenen Aufbau. Das normale MS-DOS Systemprompt können Sie mit dem internen Befehl PROMPT jederzeit individuell ändern. Weitere Angaben dazu finden Sie im Kapitel 11.

Viele Befehlseingaben erfolgen in dem allgemeinen Format:

[d:] [Weg] Funktion [Variable] **RETURN**

dabei bedeutet:

d: vor dem Programmnamen bestimmt das aufzurufende Laufwerk

Weg vor dem Programmnamen bestimmt das aufzurufende Verzeichnis in der Form:

[\] [Verzeichnis] [\Verzeichnis] [\...]

Funktion Name eines internen oder externen Befehles oder einer Batchdatei

Variable ein oder mehrere Zusatzeingaben, welche bestimmte Funktionen des Befehles aufrufen

RETURN mit dieser Taste wird die Eingabe beendet

Bemerkungen

Die Eingabe kann mit Groß- oder Kleinbuchstaben erfolgen. Es werden sämtliche Kleinbuchstaben *automatisch* vom System in Großbuchstaben umgewandelt. Werden mehrere Variable zu einem Befehl eingegeben, müssen sie durch bestimmte Zeichen getrennt werden. Dies sind: Leertaste ( ), Komma (,), Semikolon (;), Gleichheitszeichen (=) und TAB ( ).

Funktion

Die Funktion beschreibt die *auszuführende Tätigkeit* des Systems. Das kann ein interner oder externer Befehl sein, ebenso der Name einer Batchdatei. Einem externen Befehl kann zusätzlich eine Laufwerkbezeichnung und/oder der Name eines Verzeichnisses vorangestellt werden.

Tabelle 5.1. Beispiele für Funktionen interner Befehle

FUNKTION	ZWECK
DIR	zeigt eine Liste der gespeicherten Dateien (DIRectory)
PROMPT	definiert das Systemanforderungszeichen (PROMPT)
D:	definiert das Arbeitslaufwerk
TIME	zeigt und ändert die Uhrzeit

Besteht die Funktion aus einem externen Befehl, muß diesem immer der genaue Speicherplatz vorangestellt werden, falls sich das Programm nicht im aktuellen Verzeichnis befindet. Den Speicherplatz bestimmen Sie durch eine Laufwerkbezeichnung und/oder einen Weg.

Tabelle 5.2 Beispiele für Funktionen externer Befehle

FUNKTION	ZWECK
CONFIGUR	paßt das System für verschiedene Peripheriegeräte an (Programm CONFIGUR.COM ist auf dem Arbeitslaufwerk)
D:DISKCOPY	kopiert den Inhalt einer Diskette auf eine andere (Programm DISKCOPY.COM ist auf Laufwerk D)
CHKDSK	überprüft eine Diskette auf Fehler (CHecK DiSK)

Befehlszeile

Variable

In der Befehlszeile werden, durch ein Freizeichen getrennt, wahlweise *Variable* hinter der Funktion eingegeben. Diese Variable bestimmen die Funktion genauer. Geben Sie die Werte nicht selber ein, werden vom System gespeicherte Werte eingesetzt. Im Kapitel 11 werden für jeden Befehl die gespeicherten Werte näher erklärt.

Viele Befehlseingaben erfolgen in dem allgemeinen Format:

[d:] [Weg] Funktion [Variable...] **RETURN**

dabei können für [Variable...] folgende Eingaben gemacht werden:

D:	eine Laufwerkbezeichnung
Datei-name	eine zu bearbeitende Datei
Namens-zusatz	zusätzliche Eingabe für Dateiname (manchmal zwingend vorgeschrieben)
Datei-bez	eine zu bearbeitende Datei
Weg	mehrere Verzeichnisse in Folge in der Form: [\] [Verzeichnis] [\Verzeichnis...]
Buch-stabe	zusätzliche Eingabe für eine besondere Funktion, ein Schrägstrich (/) muß meistens vorangestellt werden

RETURN

Jede Eingabe in der Befehlszeile muß durch Betätigen der Taste **RETURN** beendet werden. Nur daran erkennt das System, daß die Eingabe beendet ist.

Schreibweise

Die Befehlszeile wird vom Betriebssystem (durch die Datei COMMAND.COM) Zeichen für Zeichen abgefragt und untersucht. Daher ist es erforderlich, die Eingaben nach genau festgelegten Regeln einzugeben, z. B. muß die Zeile mit einem gültigen Befehlsnamen beginnen und mit **RETURN** beendet werden. Sind in der Zeile keine Fehler entdeckt worden, wird der Befehl entsprechend ausgeführt.

Für die Eingabe der Befehlszeile gelten folgende Regeln:

- Befehle werden mit einer oder mehreren Variablen eingegeben.
- Eingaben können mit Groß- oder Kleinbuchstaben erfolgen.
- Befehle und Variable müssen durch Abgrenzungszeichen getrennt werden. Das sind Leertaste, TAB, Komma, Semikolon, Gleichzeichen und Pluszeichen.
- In der Dateibezeichnung sind nur Doppelpunkt und Punkt als Abgrenzungszeichen erlaubt, in der Form:

  D:Dateiname.Zus

- Werden Sie vom System in Dienstprogrammen aufgefordert, irgendeine Taste zu betätigen (Press any key), so kann das jede beliebige alphanumerische Taste sein.
- Geben Sie in der Befehlszeile einen zu bearbeitenden Dateinamen ein, so muß, falls vorhanden, der Namenszusatz mit eingegeben werden.
- Ein bereits eingegebener Befehl kann durch Eingabe von **CTRL-BREAK** oder **CTRL-C** (gleichzeitiges Betätigen beider Tasten) sofort abgebrochen werden.
- Reservierte Dateinamen dürfen nicht eingegeben werden.
- Zeichen zum Zusammenfassen mehrerer Dateien dürfen nicht eingegeben werden.

Befehlszeile

- Die Ausgabe auf dem Bildschirm läßt sich jederzeit durch Eingabe von **CTRL-NUM LOCK** oder **CTRL-S** anhalten. Nach Betätigen einer beliebigen Taste wird die Ausgabe fortgesetzt.

- Die Eingabe darf auch mit Hilfe der Editier- und Funktionstasten erfolgen.

- Wird die Angabe eines Ursprungslaufwerkes oder einer Ursprungsdatei (Source Drive / Source File) gefordert, so ist dies immer das Laufwerk (Datei), von der Daten übertragen werden.

- Wird die Eingabe eines Ziellaufwerkes oder einer Zieldatei (Target Disk / Target File) gefordert, so ist dies immer das Laufwerk (Datei) auf das Daten überspielt werden.

- Wird der Name einer Batchdatei eingegeben, so werden die dort gespeicherten Befehle so ausgeführt, als wären diese direkt über die Tastatur eingegeben worden.

- Bei Namensgleichheit werden von MS-DOS zu bearbeitende Programme in der Reihenfolge .COM, .EXE, .BAT aufgerufen.

Korrektur

Falsch eingegebene Befehle oder Befehlszeilen lassen sich mit Hilfe von besonderen *Tasten* schnell und problemlos korrigieren. Dafür werden von MS-DOS zwei Zwischenspeicher im RAM reserviert:

- Befehlszeilenspeicher

- Tastatureingabespeicher

Der Befehlszeilenspeicher hat eine Gesamtspeicherkapazität von 128 Zeichen. Hier wird jedes von Ihnen eingegebene Zeichen so lange gespeichert, bis Sie die Taste **RETURN** betätigen.

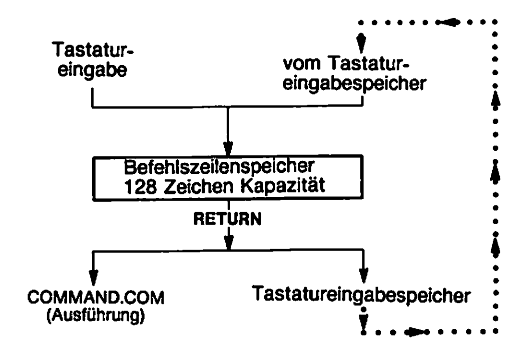


Bild 5.1
Befehlszeile und Eingabespeicher

Haben Sie die Taste **RETURN** betätigt, wird der Inhalt des Befehlszeilenspeichers komplett in den Tastatureingabespeicher kopiert. Hier wird also immer die letzte Befehlszeile gespeichert. Sie wird nur durch eine neue Eingabe überschrieben. Der Speicherinhalt steht für eine Korrektur zur Verfügung.

Mit Betätigen der Taste **RETURN** wird ebenfalls die Befehlszeile zur Datei COMMAND.COM weitergegeben. Der eingegebene Befehl kann ausgeführt werden. Ist dies nicht möglich, wird eine entsprechende Fehlermeldung ausgegeben. Sie können dann die Eingabe korrigieren.

Funktionstasten

Mit Hilfe einiger Funktionstasten lassen sich Befehlszeilen leicht ändern. Die zuletzt eingegebene Zeile kann aus dem Tastatureingabespeicher komplett oder in Teilen zurück in die Befehlszeile kopiert werden. Damit werden folgende Eingaben unterstützt:

- eine Befehlszeile kann leicht wieder eingegeben werden
- eine Befehlszeile kann leicht korrigiert werden
- eine Befehlszeile kann in leicht geänderter Form eingegeben werden

Die folgende Tabelle zeigt eine Übersicht der Funktionstasten. Eine ausführliche Beschreibung finden Sie im Kapitel 12.

Tabelle 5.3 MS-DOS Funktionstasten

TASTE	FUNKTION	BESCHREIBUNG
F1	kopiert ein Zeichen	kopiert ein Zeichen aus dem Eingabespeicher in die neue Zeile
F2x	kopiert bis Zeichen x	kopiert alle Zeichen bis zum eingegebenen (x) aus dem Eingabespeicher in die neue Zeile
F3	kopiert alle Zeichen	kopiert alle Zeichen des Eingabespeichers in die neue Zeile
F4x	überspringt bis Zeichen x	überspringt alle Zeichen des Eingabespeichers bis zum eingegebenen
F5	neuer Eingabespeicher	schreibt die neue Zeile in den Eingabespeicher
F6	beendet Eingabe	gibt ein CTRL-Z aus, Zeichen für das Ende der Datei
F7	Nullzeichen	gibt ein CTRL-@ aus, Zeichen wird mit dem Wert 00h gespeichert

Tabelle 5.3 MS-DOS Funktionstasten

TASTE	FUNKTION	BESCHREIBUNG
ALT-BREAK	Eingabe löschen	löscht den Tastaturspeicher
BACK-SPACE	Eingabe löschen	löscht ein Zeichen in der Eingabezeile und im Speicher
INS	einfügen	schaltet um auf Einfügen
DEL	löschen	löscht ein Zeichen im Eingabespeicher, nicht aber in der Eingabezeile
ESC	Eingabe löschen	löscht die gesamte Eingabezeile, nicht aber den Eingabespeicher

Beispiele

Die folgenden Beispiele zeigen, wie einfach das Arbeiten mit Hilfe der Funktionstasten ist.

Beispiel 1:

Sie wollen sich aus dem Inhaltsverzeichnis des Laufwerkes die Daten der Datei HANDBUCH.TXT anzeigen lassen. Sie geben ein:

DIR HANDBUCH.TXT **RETURN**

Der Befehl wird ausgeführt und gleichzeitig wird die Eingabe in den Tastatureingabespeicher kopiert. Mit nur zwei Tasten ist es jetzt möglich, diesen Befehl zu wiederholen. Betätigen Sie die Tasten

F3 **RETURN**

Befehlszeile

Beispiel 2

Sie wollen jetzt Informationen haben über sämtliche Dateien, die Sie unter dem Namen HANDBUCH gespeichert haben. Der Namenszusatz unterscheidet die einzelnen Dateien. In der Zeile muß dafür nur der Namenszusatz .TXT gegen den Stern (*) getauscht werden. Geben Sie ein:

F2 T

Es werden jetzt sämtliche Zeichen bis zum T ausschließlich in die Eingabezeile kopiert. Es erscheint:

DIR HANDBUCH.

Geben Sie dann den Stern (*) ein, und betätigen Sie die Taste **RETURN.**

Beispiel 3

Da es sich bei der Datei HANDBUCH.TXT um eine Textdatei handelt, können Sie den Inhalt auf den Bildschirm ausgeben. Verändern Sie die letzte Eingabe leicht, um folgenden Befehl zu erhalten:

TYPE HANDBUCH.TXT

Sie überschreiben die ersten drei Zeichen der alten Zeile (DIR) mit den Buchstaben **TYP**. Sie betätigen die Taste **INS**, fügen ein **E** hinzu, betätigen die Tasten **F2,*** und schreiben als letztes **TXT**. Nach Betätigen von **RETURN** wird der Inhalt auf dem Bildschirm angezeigt.

In der Übersicht noch einmal Ihre Eingaben:

TASTEN	FUNKTION	ERGEBNIS
T	schreibt ein Zeichen	T
Y	schreibt ein Zeichen	TY
P	schreibt ein Zeichen	TYP
INS	schaltet um auf Einfügen	
E	schreibt ein Zeichen	TYPE
F2*	kopiert Zeichen bis *	TYPE HANDBUCH.
T	schreibt ein Zeichen	TYPE HANDBUCH.T
X	schreibt ein Zeichen	TYPE HANDBUCH.TX
T	schreibt ein Zeichen	TYPE HANDBUCH.TXT
RETURN	gibt die Zeile an die Datei COMMAND.COM zum Ausführen	Inhalt wird angezeigt

Kontrolltasten

Solange Sie sich im Betriebssystem befinden, können Sie mit speziellen *Tastenkombinationen* ohne weiteren Tippaufwand einige Befehle ausführen. Die Befehle werden durch eine Kombination der Taste **CTRL** mit der entsprechend beschriebenen Taste aufgerufen. Diese Befehle sind in der Tabelle 5.4 aufgelistet.

Befehlszeile

Tabelle 5.4 MS-DOS Kontrolltasten

TASTEN	FUNKTION
ALT-ESC	schaltet das Klicken der Tastatur ein und aus
CTRL-ALT-DEL	Systemstart
CTRL-ALT-INS	kehrt zum Monitorprogramm zurück
CTRL-ALT-RETURN	unterbricht das laufende Programm und geht zum Monitorprogramm, die momentanen Registerinhalte werden angezeigt
CTRL-BREAK	unterbricht sofort die Abarbeitung eines Befehls
CTRL-C	unterbricht sofort die Abarbeitung eines Befehls
CTRL-H	löscht das zuletzt eingegebene Zeichen
CTRL-I	bewegt den Cursor zum nächsten Tab
CRTL-J	fügt ein physikalisches Zeilenende ein, macht einen Zeilenvorschub
CTRL-NUM LCK	unterbricht die Ausgabe auf dem Bildschirm, wird nach Betätigen einer Taste fortgesetzt
CTRL-P	gibt die Bildschirmausgabe gleichzeitig auf den Drucker, wird durch erneutes Betätigen beendet
CTRL-PRTSC	gibt die Bildschirmausgabe gleichzeitig auf den Drucker, wird durch erneutes Betätigen beendet
CRTL-RETURN	fügt ein physikalisches Zeilenende ein, macht einen Zeilenvorschub
CTRL-S	unterbricht die Ausgabe auf dem Bildschirm, wird nach Betätigen einer Taste fortgesetzt
CTRL-U	löscht die gesamte Eingabezeile
CTRL-W	löscht das momentane Wort
CTRL-Z	gibt ein ^Z aus, das Zeichen für Dateiende
SHIFT-PRTSC	kopiert den Bildschirminhalt komplett auf den Drucker

Grundzüge MS - DOS

Automatische Befehlseingabe

In einer *Batchdatei* werden nacheinander mehrere Befehle auf Betriebssystemebene abgearbeitet. Bei einer direkten Eingabe müßten diese Befehle hintereinander eingegeben werden. Werden diese Befehle vorher in eine Batchdatei geschrieben, werden sie so abgearbeitet, als wenn sie wiederum direkt eingegeben werden. Damit ist es nicht erforderlich, wiederkehrende Routinen jedesmal neu zu schreiben.

MS-DOS erkennt eine Batchdatei am Namenszusatz *.BAT*. Bei der Erzeugung einer Batchdatei müssen Sie beachten, daß der Dateiname nicht schon für eine .COM- oder .EXE-Datei auf der Diskette vorhanden ist. Die Batchdatei wird ohne Namenszusatz aufgerufen. MS-DOS sucht auf der Diskette erst nach Dateien mit den Zusätzen .COM oder .EXE und würde deshalb diese Batchdatei nie finden.

Beim Systemstart wird von der Datei COMMAND.COM automatisch nach einer Batchdatei mit dem Namen *AUTOEXEC.BAT* gesucht. Wird diese Datei gefunden, werden sofort die dort gespeicherten Befehle abgearbeitet. Es entfallen die Anfragen für Datum und Zeit.

Die Datei AUTOEXEC.BAT kann von Ihnen selber geschrieben werden.

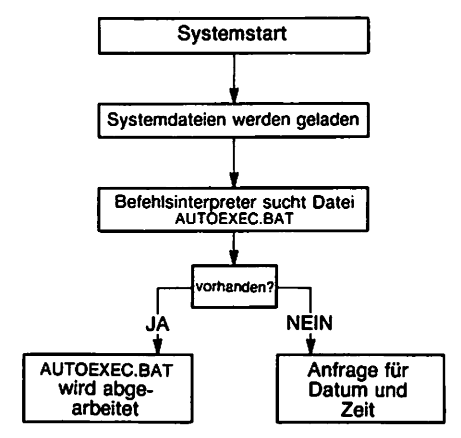


Bild 5.2
MS-DOS und die Datei AUTOEXEC.BAT

Automatische Befehlseingabe

Die Datei AUTOEXEC.BAT

Im Inhaltsverzeichnis, das die Systemdateien enthält, muß die von Ihnen geschriebene Datei *AUTOEXEC.BAT* vorhanden sein. Die Datei können Sie Ihren Wünschen entsprechend selber schreiben.

Ein Beispiel für eine AUTOEXEC-Datei:

Schritt 1: Geben Sie folgende Zeile direkt hinter dem Prompt ein:

COPY CON AUTOEXEC.BAT **(RETURN)**

Sie schreiben damit alle folgenden Zeichen von der Tastatur aus direkt in die Datei AUTOEXEC.BAT.

Schritt 2: Geben Sie jetzt folgende Befehle zeilenweise ein. Schließen Sie jede Zeile mit einem **RETURN** ab.

DATE	(Datumsabfrage)
TIME	(Zeitabfrage)
KEYBGR	(deutschen Zeichensatz laden)
C:	(Laufwerk wechseln)
BASICA	(BASIC-Interpreter aufrufen)

Schritt 3: Beenden Sie die Eingabe in die Datei mit dem Zeichen für Dateiende: **^Z**. Betätigen Sie die Tasten **F6** und **RETURN**. Es erscheint dann die Meldung:

1 File(s) copied   1 Datei kopiert

Zur Kontrolle können Sie jetzt im Inhaltsverzeichnis nachsehen, ob ein entsprechender Eintrag für die Datei vorhanden ist. Geben Sie dazu ein:

DIR

Automatische Befehlsverarbeitung

Bei intensivem Gebrauch des Betriebssystemes MS-DOS wird es vielfach vorkommen, daß Sie Befehle in derselben Reihenfolge benutzen. Es ist nicht erforderlich, diese Befehle immer wieder neu einzugeben. Diese Befehle können in einer *Batchdatei* gespeichert werden. Dies ist eine Datei zur automatischen Befehlsverarbeitung. Das Betriebssystem holt sich selbständig die dort gespeicherten Befehle in der eingegebenen Reihenfolge. Dies ist zu vergleichen mit einem *Stapel*, der schrittweise abgetragen wird. Daher wird eine Batchdatei auch eine *Stapeldatei* genannt.

Erstellen

Eine Batchdatei kann mit dem MS-DOS Dienstprogramm EDLIN oder jedem anderen Textverarbeitungsprogramm erstellt werden. Ebenso ist es möglich, diese Datei von der Tastatur aus direkt einzugeben. Geben Sie dazu den Befehl ein:

```
COPY CON BEISPIEL.BAT
```

Sie schreiben (Copy = kopieren) jetzt von der Tastatur (Con) aus direkt in die Datei. Abgeschlossen wird die Datei mit dem Zeichen ^Z (Taste **F6** oder **CTRL-Z**).

Ausführen

Eine Batchdatei wird nur mit ihrem Dateinamen aufgerufen. Die Eingabe des Zusatzes .BAT ist nicht erforderlich.

Variable, die nach dem Dateinamen eingegeben werden, müssen durch ein Freizeichen getrennt werden.

Automatische Befehlsverarbeitung

Die Diskette mit der Batchdatei darf während der Abarbeitung nicht durch eine andere Diskette ersetzt werden. Dann ist es nicht möglich, mit der Abarbeitung fortzufahren, und es erscheint die Meldung:

Batch file missing	Batchdatei fehlt
Insert disk with batch file and press any key when ready	Legen Sie die Diskette mit der Batchdatei wieder ein. Betätigen Sie eine Taste, wenn fertig

Wird während des Abarbeitens die Tastenkombination **CTRL-C** oder **CTRL-BREAK** betätigt, wird die Abarbeitung angehalten und es erscheint die Frage:

Terminate batch job (Y/N)?	Batchdatei beenden (J/N)?

Geben Sie ein **N**(ein) ein, wird der laufende Befehl übersprungen und die Abarbeitung mit dem folgenden Befehl wieder aufgenommen. Geben Sie ein **Y** für Ja ein, verlassen Sie die Batchdatei und es erscheint wieder das Prompt des Betriebssystemes.

Als letzten Befehl in einer Batchdatei können Sie den Namen einer weiteren Batchdatei eingeben. Diese Datei wird dann aufgerufen und ebenfalls abgearbeitet.

Befehle

Prinzipiell ist es möglich, jeden internen und externen Befehl in einer Batchdatei zu verarbeiten. Einige der internen Befehle sind speziell für die Abarbeitung in einer Batchdatei vorgesehen. DieTabelle 5.5 zeigt eine Übersicht. Nähere Einzelheiten zu diesen Befehlen finden Sie im Kapitel 11.

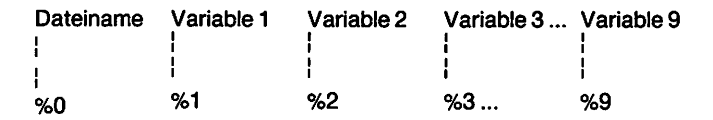
Tabelle 5.5 Befehle für Batchdateien

BEFEHL	FUNKTION
ECHO	zeigt Befehle der Batchdatei auf dem Bildschirm an bzw. unterdrückt die Ausgabe
FOR	veranlaßt die wiederholte Abarbeitung des Befehls mit verschiedenen Variablen
GOTO	unterbricht die kontinuierliche Abarbeitung und springt zur bezeichneten Zeile
IF	macht die Abarbeitung eines Befehls von der Erfüllung einer Bedingung abhängig
PAUSE	unterbricht die Abarbeitung einer Batchdatei und zeigt einen Text an
REM	zeigt während der Abarbeitung einer Batchdatei Kommentare auf dem Bildschirm an
SHIFT	ermöglicht die Abarbeitung von mehr als 10 Variablen

Variable

Damit eine einmal geschriebene Batchdatei flexibel eingesetzt werden kann, ist es möglich, *Variable* zu verarbeiten. Es kann eine beliebige Anzahl von Variablen eingesetzt werden (unter MS-DOS 1.25 sind nur 10 Variable möglich). Die Variablen werden in der Datei mit den Ziffern 0 bis 9 (gleich 10 Variable) und einem vorangestellten Prozentzeichen (%) bezeichnet. Die entsprechenden Werte dafür werden in der Befehlszeile hinter dem Dateinamen eingegeben. Jeder Wert wird durch ein Freizeichen getrennt. Dabei werden die eingegebenen Werte in ihrer Reihenfolge in der Datei ersetzt. Der erste zu ersetzende Wert ist der Dateiname selber (%0). Dadurch ist es möglich, daß sich die Datei selber wieder aufruft. Ein Beispiel:

Automatische Befehlsverarbeitung



Sind mehr als 10 Variable in der Befehlszeile eingegeben, werden Sie in der Datei nur dann übernommen, wenn mit dem Befehl SHIFT alle Variablen um eine Stelle nach links gesetzt werden. So kann aus der 11. Variablen die 10. Variable (%9) werden.

Ein Beispiel:

Es ist eine Batchdatei mit dem Namen SICHERN vorhanden, die folgenden Inhalt hat:

COPY %1 A:%2.NEU
TYPE %1
PAUSE
TYPE %2.NEU
REM Dies ist die Batchdatei
TYPE %0

Sie geben ein:

SICHERN C:TEXT.ASC BRIEF

Die Variablen in den einzelnen Zeilen werden jetzt durch Ihre Eingaben ersetzt. Die Befehle werden folgendermaßen abgearbeitet:

COPY C:TEXT.ASC A:BRIEF.NEU
TYPE C:TEXT.ASC
PAUSE
TYPE A:BRIEF.NEU
REM Dies ist die Batchdatei
TYPE SICHERN.BAT

Automatische Befehlsverarbeitung

Ihre Eingabe in der Befehlszeile, mit der Sie die Batchdatei aufrufen, wird in Variable zerlegt. Die Variablen werden in der Datei folgendermaßen eingesetzt:

1. VARIABLE (%0)	2. VARIABLE (%1)	3. VARIABLE (%2)
SICHERN	C:TEXT.ASC	BRIEF

Beispiele

Die folgenden fünf Beispiele zeigen verschiedene Möglichkeiten für Batchdateien.

Beispiel 1

Die erste Batchdatei formatiert und überprüft neue Disketten. Geben Sie ein:

```
A>COPY CON NEUDISK.BAT
REM Diese Datei heißt NEUDISK.BAT
REM Sie formatiert und überprüft neue Disketten
PAUSE Legen Sie eine neue Diskette in Laufwerk B:.
FORMAT B:/S
CHKDSK B:
DIR B:
```

Nachdem Sie die Eingabe mit CTRL-Z (Taste **F6**) abgeschlossen haben, erscheint die Meldung:

```
1 File(s) copied                1 Datei kopiert
```

Diese Datei können Sie jetzt durch Eingabe von NEUDISK aufrufen.

Automatische Befehlsverarbeitung

Beispiel 2

Diese Batchdatei übersetzt (assembliert) Quellprogramme in Maschinensprache. Geben Sie ein:

```
A> COPY CON ASSEMBLE.BAT
REM Diese Datei assembliert Quellcodes
ECHO Diese Befehle werden ausgeführt:
TYPE 0%.BAT
COPY %1.TXT %2.ASM
MASM %2,%2,%2
```

Beenden Sie auch hier wieder die Eingabe mit **CRTL-Z**. Rufen Sie die Datei mit der folgenden Zeile auf:

```
ASSEMBLE PROG SPIEL
```

Die Befehle werden so abgearbeitet, als würden Sie folgendes über die Tastatur eingeben:

```
REM Diese Datei assembliert Quellcodes
ECHO Diese Befehle werden ausgeführt:
TYPE ASSEMBLE.BAT
COPY PROG.TXT SPIEL.ASM
MASM SPIEL, SPIEL, SPIEL
```

Beispiel 3

Die folgende Batchdatei überprüft das Inhaltsverzeichnis Ihrer Diskette auf das Vorhandensein der angegebenen Datei. Geben Sie folgendes ein:

```
A> COPY CON SUCHE.BAT
ECHO OFF
IF EXIST %1 GOTO X
ECHO Die Datei ist nicht vorhanden
GOTO END
:X
ECHO Die Datei ist vorhanden
DIR %1
:END
```

Beispiel 4

Die folgende Batchdatei zeigt den Inhalt von Textdateien auf dem Bildschirm an. Geben Sie ein:

```
A>COPY CON ZEIGE.BAT
FOR %%A IN (%1) DO TYPE %%A
```

Diese Datei kann mit verschiedenen Variablen zusammen aufgerufen werden. Geben Sie nur einen Dateinamen ein, wird der Inhalt dieser Datei angezeigt. Ein Beispiel

```
ZEIGE BRIEF.TXT
```

zeigt nur diese eine Datei an. Geben Sie aber ein

```
ZEIGE *.BAT
```

wird der Inhalt sämtlicher Batchdateien angezeigt.

Beispiel 5

Die folgende Batchdatei löscht eine oder mehrere angegebene Dateien. Geben Sie ein:

```
A>COPY CON LOESCHEN.BAT
ECHO OFF
:LOOP
IF "%1"==""GOTO DONE
DEL %1
SHIFT
GOTO LOOP
:DONE
ECHO Datei(en) gelöscht
```

Diese Datei kann auch wie das Beispiel 4 auf zwei verschiedene Arten aufgerufen werden.

Es können hinter dem Batchnamen mehrere konkrete Dateinamen zum Löschen eingegeben werden. Diese werden dann durch den Befehl SHIFT jeweils an die Stelle der Variablen %1 gesetzt.

Geben Sie hinter dem Batchnamen einen Dateinamen mit Stern ein, so werden sämtliche Dateien, die so erfaßt werden, gelöscht.

Hinweise für Benutzer mit einem Laufwerk

Bei Geräten mit nur *einem Laufwerk* müssen einige Befehle unterschiedlich gehandhabt werden. Ihre Schreibweise ist gleich. Da nur ein Laufwerk vorhanden ist, ist dieses gleichzeitig *Laufwerk A:* und *Laufwerk B:*. Jetzt greift Laufwerk A: mit *Diskette A:* auf Laufwerk A: mit *Diskette B:* zu. Das Prompt ist immer A>. Sie werden vom System aufgefordert, die Disketten zu vertauschen. Es erscheint dann eine der beiden Meldungen:

Place disk A in drive A: Press any key when ready.	Legen Sie Diskette A in Laufwerk A: Drücken Sie eine Taste, wenn fertig
Place disk B in drive A: Press any key when ready.	Legen Sie Diskette B in Laufwerk A: Drücken Sie eine Taste, wenn fertig

Diese Meldungen werden bei den Befehlen ausgegeben, die auf unterschiedliche Laufwerke zugreifen können. Hier eine Aufstellung der Befehle:

CHKDSK	COPY
DEL oder ERASE	DIR
DISKCOMP	DISKCOPY
FC	FORMAT
RECOVER	REN
TYPE	

Benutzen Sie einen der Befehle in einer Batchdatei, so wird die Abarbeitung angehalten, und Sie werden aufgefordert, die Diskette zu wechseln. Ein Beispiel:

COPY COMMAND.COM B:

Nach Eingabe dieses Befehls erscheint folgende Meldung:

Place disk B in drive A: Press any key when ready.	Legen Sie Diskette B in Laufwerk A: Drücken Sie eine Taste, wenn fertig

Nach Betätigen irgendeiner beliebigen Taste wird die Datei kopiert, und es erscheint die Meldung:

1 File(s) copied	1 Datei kopiert

Befehlsunterbrechung

Es gibt zwei Arten der Befehlsunterbrechung.

Der Anwender, also Sie, *unterbricht absichtlich* die Abarbeitung eines Programmes. Nach Betätigen der beiden Tasten **CTRL-BREAK** wird das Programm sofort abgebrochen, und es erscheint wieder das Prompt. Die Tastenkombination **CTRL-C** arbeitet in ähnlicher Weise. Geben Sie den Befehl **BREAK ON** ein, so wird mit der Kombination **CTRL-C** das Programm ebenfalls sofort abgebrochen. Geben Sie ein **BREAK OFF** ein, so wird mit der Kombination **CTRL-C** das Programm nur dann abgebrochen, wenn eine Ausgabe zu einem Peripheriegerät erfolgt. Der Abbruch durch **CTRL-BREAK** erfolgt immer sofort, unabhängig vom Befehl BREAK. Diese Tastenkombinationen können in Anwendungsprogrammen andere Funktionen aufrufen.

Das Programm kann ebenfalls abgebrochen werden, wenn ein Fehler auftritt. Dies wird dann durch eine entsprechende Fehlermeldung angezeigt.

Fehlermeldungen

Kann das Betriebssystem *Befehle* nicht ausführen, so wird eine entsprechende Fehlermeldung ausgegeben. Im folgenden Abschnitt werden alle Fehlermeldungen des Betriebssystemes (ohne Dienstprogramme) in alphabetischer Reihenfolge gezeigt. Neben dem englischen Fehlertext sehen Sie eine deutsche Übersetzung. Der folgende kurze Abschnitt beschreibt den Fehler und gibt Hinweise zur Beseitigung.

Art ERROR WHILE	Fehler während der
i/o-action ON DRIVE X	Ein-/Ausgabe auf Laufwerk X
Abort, Retry, Ignore:	Abbrechen, Wiederholen, Ignorieren

Dabei kann die Art des Fehlers folgende sein:

Write protect	Schreibschutzfehler
Bad unit	Falsche Einheit
Not ready	Nicht bereit
Bad command	Falscher Befehl
Data	Datenfehler
Bad call format	Falscher Aufruf
Seek	Suchfehler
Non-DOS disk	Keine DOS-Diskette
Sector not found	Sektor nicht gefunden
No paper	Kein Papier
Write fault	Schreibfehler
Read fault	Lesefehler
General Failure	Allgemeiner Fehler
Sharing Violation	Datei gesperrt
Lock Violation	Datensatz gesperrt
Invalid Disk Change	Nicht erlaubter Diskettenwechsel
FCB unavailable	FCB nicht mehr vorhanden

Die Ein-/Ausgabe bezieht sich auf das Lesen (READING) oder Schreiben (WRITING). Mit X wird das Laufwerk bezeichnet, das den Fehler verursacht hat.

ERKLÄRUNG:
Tritt ein Laufwerkfehler während des Programmes oder Befehles auf, versucht MS-DOS fünfmal auf das bezeichnete Laufwerk zuzugreifen. Gelingt dies nicht, erscheint eine der oben beschriebenen Fehlermeldungen. Darunter steht immer der Satz Abort, Retry, Ignore. Damit haben Sie drei Möglichenkeiten, auf den Fehler zu reagieren:

A	Abort	Durch Eingabe eines **A** brechen Sie den Vorgang ab.
I	Ignore	Durch Eingabe eines **I** weisen Sie das System an, den fehlerhaften Sektor zu ignorieren und fortzufahren.
R	Retry	Nach Eingabe eines **R** wird der Vorgang wiederholt. Dies ist möglich, wenn das Laufwerk nicht geschlossen war oder die Diskette schreibgeschützt ist.

Diese Fehlermeldungen werden meistens mit **R** (wiederholen) oder **A** (abbrechen) beantwortet.

Art ERROR WHILE	Fehler während der
i/o-action ON DRIVE X	Ein-/Ausgabe auf Laufwerk X
Sector adress of error	Adresse des fehlerhaften Sektors
is *nnnn*	ist *nnnn*
Abort, Retry, Ignore:	Abbrechen, Wiederholen, Ignorieren

ERKLÄRUNG:
Findet das Betriebssystem auf der Winchester einen Fehler, ändert sich die Fehlermeldung gegenüber einem Diskettenfehler. Es wird immer die Adresse des fehlerhaften Sektors mit angezeigt. Die Adresse des fehlerhaften Sektors wird hexadezimal (*nnnn*) angegeben. Die logische Adresse des gesamten Winchesterlaufwerkes beginnt mit dem ersten absoluten Sektor bei 0000h.

Versuchen Sie, diesen Fehler mit Eingabe von **R** zu beheben. Tritt dieser Fehler öfter auf oder kann er nicht behoben werden, sollten Sie erst die Daten der Winchester komplett sichern. Danach überprüfen Sie das Laufwerk, das den Sektor sperrt, mit dem Dienstprogramm DETECT. Formatieren Sie die Winchester neu, und kopieren Sie mit dem Dienstprogramm RESTORE Ihre Daten zurück.

Fehlermeldungen

File allocation table bad for drive X	Dateiordnungstabelle beschädigt im Laufwerk X

ERKLÄRUNG:
Die Kopie des FILE ALLOCATION TABLE (FAT), die im Arbeitsspeicher vorhanden ist, weist auf nicht vorhandene Sektoren hin. Das ist möglich, wenn die Diskette nicht formatiert oder beschädigt ist. Starten Sie das System neu, und überprüfen Sie die Diskette mit dem Befehl CHKDSK ohne Eingabe von /F. Ist die Diskette be-schädigt, erscheint jetzt eine entsprechende Meldung (s. a. Kapitel 11).

Access denied	Zugriff nicht möglich

ERKLÄRUNG:
Sie haben versucht, eine Datei zu verändern (schreiben), die als Nur-Lese-Datei gekennzeichnet ist. Ändern Sie mit dem Programm ATTRIB den Status der Datei.

Bad call format	Falscher Aufruf

ERKLÄRUNG:
Die einem Gerätetreiber (Programm zum Anschluß eines Peripheriegerätes) übergebenen Variablen sind falsch. Benutzen Sie einen anderen Gerätetreiber, oder beheben Sie (lassen Sie vom Verkäufer beheben) den Programmfehler des Treibers. Ist dies ein Treiber von Zenith Data Systems, wenden Sie sich an Ihren zuständigen Händler.

Bad command or filename	Befehl oder Datei falsch

ERKLÄRUNG:
Der eingegebene Befehl wurde falsch geschrieben, oder die Datei ist auf dem Laufwerk nicht vorhanden.

Bad or missing Command Interpreter	Datei COMMAND.COM fehlt oder ist falsch

ERKLÄRUNG:
Der Befehlsinterpreter des Betriebssystemes fehlt auf dem Startlaufwerk. Es kann nicht weiter gebootet werden. Das kann die Datei COMMAND.COM sein oder aber ein anderer Interpreter, der durch die Datei CONFIG.SYS aufgerufen wird. Kopieren Sie die richtige Datei auf das Startlaufwerk, und booten Sie erneut.

Bad unit	Falsche Einheit

ERKLÄRUNG:
Die einem Gerätetreiber übergebenen Variablen sind falsch. Benutzen Sie einen anderen Gerätetreiber, oder beheben Sie (lassen Sie vom Verkäufer beheben) den Programmfehler des Treibers. Ist dies ein Treiber von Zenith Data Systems, wenden Sie sich an Ihren zuständigen Händler.

Batch file missing	Batchdatei fehlt
Insert disk with batch file and press any key when ready	Legen Sie die Diskette mit der Batchdatei wieder ein. Betätigen Sie eine Taste, wenn fertig.

ERKLÄRUNG:
Sie haben die Diskette mit der Batchdatei entfernt, die gerade abgearbeitet wurde. Legen Sie die Diskette wieder in das Laufwerk ein.

Cannot do binary reads from a device	Es können keine binären Daten vom Peripheriegerät gelesen werden.

ERKLÄRUNG:
MS-DOS benötigt eine End-of-File Markierung (1Ah), wenn es Daten liest. Das angeschlossene Peripheriegerät sendet aber dieses Zeichen nicht. Das Ende der Übertragung wird durch ein Protokoll definiert. Sie benötigen für dieses Gerät ein Programm, das dieses Protokoll liest.

Fehlermeldungen

Cannot load COMMAND, system halted

Die Datei COMMAND.COM kann nicht geladen werden. Das System wird angehalten.

ERKLÄRUNG:
Die Systemdateien sind im Arbeitsspeicher verändert worden und arbeiten nicht mehr fehlerfrei. Nach Beendigung eines Anwendungsprogrammes kann daher der Befehlsinterpreter nicht mehr nachgeladen werden. Starten Sie das System neu.

Cannot start COMMAND, exiting

Die Datei COMMAND.COM kann nicht aufgerufen werden. Der Vorgang wird abgebrochen.

ERKLÄRUNG:
Aus einem Anwendungsprogramm heraus wird versucht, mit Hilfe der Datei COMMAND.COM einen internen Befehl auszuführen oder ein weiteres Programm nachzuladen. Dies ist jedoch nicht möglich, daher wird der Vorgang abgebrochen.

Content of destination lost before copy

Inhalt der Kette vor dem Kopieren verloren

ERKLÄRUNG:
Dieser Fehler tritt auf, wenn mit einem Kopierbefehl mehrere Dateien zusammenkopiert werden. Benennen Sie die neue Datei dabei mit einem Namen der zu kopierenden Dateien, so tritt dieser Fehler während des Kopiervorganges auf. Die betreffende Datei ist dann zerstört (s. a. Kapitel 11).

Current drive is no longer valid

Aktuelles Laufwerk ist nicht mehr gültig

ERKLÄRUNG:
Nach mehrfachen Plattenzugriffen hat die Datei IO.SYS immer den gleichen Fehler gemeldet, z. B. versuchen Sie eine Diskette mit 48-tpi in einem 1,2 Mb Laufwerk zu benutzen. Das Betriebssystem sperrt das Laufwerk damit für alle weiteren Zugriffe.

Fehlermeldungen

Data error reading/writing drive X	Datenfehler beim Lesen/Schreiben des Laufwerkes X

ERKLÄRUNG:
Das Betriebssystem kann die Diskette nicht fehlerfrei bearbeiten. Die Diskette ist höchstwahrscheinlich beschädigt. Versuchen Sie den Fehler mit Eingabe von **R**(etry) zu beheben.

Duplicate filename or File not found	Dateiname doppelt oder Datei nicht gefunden

ERKLÄRUNG:
Dieser Fehler tritt auf, wenn Sie mit dem Befehl RENAME eine Datei umbenennen und die Datei nicht vorhanden ist. Ebenfalls, wenn der neue Dateiname schon auf der Diskette vorhanden ist.

Error in EXE file	Fehler der EXE-Datei

ERKLÄRUNG:
Das Lademodul der EXE-Datei ist fehlerhaft. Das Betriebssystem kann diese Datei nicht mehr laden. Benutzen Sie eine fehlerfreie Kopie des Programmes.

Error writing to device	Schreibfehler des Peripheriegerätes

ERKLÄRUNG:
Sie versuchen Daten direkt auf ein Peripheriegerät zu geben. Das Gerät arbeitet nicht fehlerfrei, z. B. ist der Drucker nicht eingeschaltet oder hat kein Papier mehr. Überprüfen Sie das Gerät.

Fehlermeldungen

EXEC failure	Fehler durch Systemanruf

ERKLÄRUNG:
Diese Fehlermeldung kann durch eine der folgenden Ursachen hervorgerufen werden:

- Die aufgerufene Datei existiert nicht in der Form, in der sie aufgerufen wurde.
- Das aufgerufene .EXE-Programm enthält im Kopf nicht die von MS-DOS geforderten Informationen.
- Der Arbeitsspeicher ist zu klein.
- Durch die Datei CONFIG.SYS wurden Systemerweiterungsprogramme geladen. Deshalb ist der Arbeitsspeicher zu klein.
- Es sind Anpassungsprogramme zusätzlich geladen worden. Der Arbeitsspeicher ist zu klein.
- Es wurde ein falscher Funktionsaufruf benutzt. Funktion Nummer 0, 1 oder 3 ist richtig.

FCB unavailable	FCB nicht mehr vorhanden

ERKLÄRUNG:
Durch eine Netzwerkfunktion hat MS-DOS den letzten geöffneten FCB geschlossen, da der Speicherplatz benötigt wurde. Ihr Programm arbeitet jedoch mit diesem FCB (File Control Block). Ändern Sie mit dem CONFIG.SYS-Befehl FCBS die Zahl der erlaubten FCBs.

File cannot be copied onto itself 0 File(s) copied	Datei kann nicht auf sich selber kopiert werden. Keine Datei kopiert.

ERKLÄRUNG:
Die Datei soll mit dem Befehl COPY in sich selber kopiert werden. Das ist nicht möglich.

File creation error	Falsche Dateianlage

ERKLÄRUNG:
Im Inhaltsverzeichnis der Diskette oder des Winchesterlaufwerkteiles ist kein Platz mehr, um eine weitere Datei zu speichern. Sofern noch freier Speicherplatz auf der Platte vorhanden ist, müssen im Verzeichnis Einträge gelöscht werden, damit eine weitere Datei gespeichert werden kann.

File not found	Datei nicht gefunden

ERKLÄRUNG:
Sie haben eine nicht vorhandene Datei benannt. Überprüfen Sie Ihre Eingabe.

FOR cannot be nested	FOR darf nicht verschachtelt werden

ERKLÄRUNG:
Sie rufen mit dem Befehl FOR erneut den Befehl FOR auf. Diese Verschachtelung ist im Betriebssystem nicht erlaubt.

General failure reading/writing drive x	Allgemeiner Fehler beim Lesen/Schreiben des Laufwerkes x.

ERKLÄRUNG:
Diese Fehlermeldung wird durch ein fehlerhaftes Peripheriegerät verursacht. Können Sie keinen besonderen Grund dafür finden, ziehen Sie einen Programmierer hinzu.

Insufficient disk space	Nicht genug Speicherkapazität auf der Platte.

ERKLÄRUNG:
Die Speicherkapazität der Platte, auf die geschrieben werden soll, reicht nicht aus für die Datei. Wechseln Sie die Diskette (Teilbereich der Winchester)

Fehlermeldungen

Insufficient memory	Arbeitsspeicher zu klein

ERKLÄRUNG:
Der Befehlsinterpreter COMMAND.COM benötigt einen größeren Arbeitsspeicher zur Ausführung des Befehles. Der freie Speicher ist zu klein, weil:

- durch die Datei CONFIG.SYS Systemerweiterungsprogramme geladen wurden,

- Anpassungsprogramme zusätzlich geladen wurden.

Intermediate file error during pipe	Dateifehler während der Befehlsverkettung

ERKLÄRUNG:
Der Speicherplatz des Arbeitslaufwerkes ist voll belegt. Während der Befehlsverkettung werden vom System zwei Arbeitsdateien auf die Platte geschrieben (s. a. Kapitel 8). Dies ist jetzt nicht möglich. Wechseln Sie die Diskette (Teilbereich der Winchester).

Invalid COMMAND.COM	Datei COMMAND.COM fehlerhaft

ERKLÄRUNG:
Das Betriebssystem versucht den Befehlsinterpreter nachzuladen. Da Sie mit der vorhandenen Version jedoch nicht gebootet haben, stimmen die Kontrollwerte nicht überein. Legen Sie die Startdiskette wieder ein, oder booten Sie neu.

Invalid date Enter new date:	Datum falsch Geben Sie ein neues Datum ein:

ERKLÄRUNG:
Das Datum wurde falsch geschrieben, oder es wurden falsche Werte eingegeben. Geben Sie das Datum erneut ein.

Invalid device — Falsches Peripheriegerät

ERKLÄRUNG:
Sie wollen auf ein Peripheriegerät zugreifen, das von MS-DOS nicht unterstützt wird, oder das bezeichnete Gerät ist nicht angeschlossen.

Invalid disk change reading/writing drive x — Nicht erlaubter Diskettenwechsel während Lesen/Schreiben des Laufwerkes x

ERKLÄRUNG:
Sie haben die gerade bearbeitete Diskette gewechselt. Legen Sie die vorherige Diskette wieder in das Laufwerk.

Invalid directory — Falsches Inhaltsverzeichnis

ERKLÄRUNG:
Das angegebene Verzeichnis ist nicht vorhanden. Überprüfen Sie Ihre Eingabe.

Invalid drive in search path — Falsche Laufwerkbezeichnung im Weg

ERKLÄRUNG:
Sie haben mit dem Befehl PATH ein nicht vorhandenes Laufwerk bestimmt.

Invalid drive specification — Falsche Laufwerkbezeichnung

ERKLÄRUNG:
Sie haben eine ungültige Laufwerkbezeichnung eingegeben. Überprüfen Sie Ihre Eingabe.

Invalid file name or file not found — Falscher Dateiname oder Datei nicht vorhanden

ERKLÄRUNG:
Der Dateiname enthält ungültige Zeichen (s. a. Kapitel 1). Überprüfen Sie Ihre Eingabe.

Fehlermeldungen

Invalid number of parameters	Falsche Anzahl von Zusatzeingaben

ERKLÄRUNG:
Sie haben beim Befehlsaufruf zu viele oder zu wenige Zusatzeingaben vorgenommen. Überprüfen Sie Ihre Eingabe.

Invalid parameter	Falsche Zusatzeingabe

ERKLÄRUNG:
Eine der eingegebenen Zusatzeingaben der Befehlszeile war falsch. Überprüfen Sie Ihre Eingabe

Invalid path, not directory, or directory not empty	Falscher Weg, kein Verzeichnis oder Verzeichnis nicht leer.

ERKLÄRUNG:
Der Name des Inhaltsverzeichnisses war falsch oder es wurde versucht, ein Verzeichnis zu löschen, in dem noch Dateien eingetragen waren. Überprüfen Sie Ihre Eingabe.

Invalid path or file name	Weg oder Dateiname falsch

ERKLÄRUNG:
Das eingegebene Verzeichnis oder der Dateiname war falsch. Überprüfen Sie Ihre Eingabe.

Invalid time Enter new time:	Falsche Zeit Geben Sie eine neue Zeit ein:

ERKLÄRUNG:
Die Zeit wurde falsch geschrieben, oder es wurden falsche Werte eingegeben. Geben Sie die Zeit erneut ein.

Label not found	Bezeichnung nicht vorhanden

ERKLÄRUNG:
Die hinter dem Befehl GOTO genannte Bezeichnung ist in der Batchdatei nicht vorhanden.

Lock-Violation reading/writing drive x	Datensatz gesperrt während Lesen/Schreiben des Laufwerkes x

ERKLÄRUNG:
Ihr Programm versucht auf einen Datensatz eines Programmes zuzugreifen, der von einem anderen Programm bereits gesperrt wurde. Warten Sie einen Moment (bis der Satz wieder freigegeben ist), und wiederholen Sie dann mit **R**(etry), oder brechen Sie mit **A**(bort) ab.

Memory allocation error	Fehler bei Speicherzuweisung

ERKLÄRUNG:
Das Betriebssystem kann Ihrem Programm den geforderten Speicher nicht zuweisen. Diese Fehlermeldung erscheint nur, wenn die interne Speichertabelle von DOS beschädigt ist.

No free file handles	Keine Dateinummer mehr frei

ERKLÄRUNG:
Ihr Programm benötigt eine weitere Dateinummer für eine Datei. Alle möglichen Nummern sind bereits vergeben. Bestimmen Sie mit dem Befehl FILES der Datei CONFIG.SYS eine größere Dateizahl, und starten Sie das System neu.

Non-DOS disk reading/writing drive x	Keine DOS-Diskette während Lesen/Schreiben des Laufwerkes x

ERKLÄRUNG:
Die zu bearbeitende Diskette ist noch nicht, oder von einem anderen Betriebssystem formatiert worden.

Not ready drive x reading/writing drive x	Laufwerk x nicht bereit während Lesen/Schreiben des Laufwerkes x

ERKLÄRUNG:
Das bezeichnete Laufwerk ist nicht geschlossen, oder es fehlt die Diskette.

Fehlermeldungen

No paper writing device x	Kein Papier, Fehler beim Schreiben des Gerätes x

ERKLÄRUNG:
Der Systemdrucker hat kein Papier mehr.

No System	Kein System

ERKLÄRUNG:
Es wird versucht, das System von einer Platte zu starten, die zwar den Boot-Loader, aber keine Systemdateien enthält. Der Boot-Loader sucht die Systemdateien an einem genau definierten Platz. Findet er sie dort nicht, erscheint diese Fehlermeldung.

Out of environment space	Speicher für die System- variablen belegt

ERKLÄRUNG:
In einem speziellen Teil des RAM, der Umgebung, werden Variable gespeichert, die vom Betriebssystem oder Anwenderprogrammen zur Verarbeitung benötigt werden. Dieser Speicher hat eine Kapazität von 200 Bytes. Soll eine weitere Variable dort abgelegt werden, und der Speicher ist voll, erscheint diese Fehlermeldung (s. a. Kapitel 11 Befehl SET).

Programm too big to fit in memory	Das Programm ist für den Speicher zu groß

ERKLÄRUNG:
Das aufgerufene Programm ist für den noch freien Arbeitsspeicher zu groß. Sie benötigen für dieses Programm einen größeren Speicher, oder Sie müssen Systemerweiterungen und resident gespeicherte Programme löschen.

Read fault drive x reading/writing drive x	Lesefehler des Laufwerkes x während Lesen/Schreiben

ERKLÄRUNG:
Die Diskette des bezeichneten Laufwerkes ist beschädigt. Prüfen Sie mit CHKDSK, und nehmen Sie eventuell eine andere Diskette.

Sector not found Abort, Ignore, Retry:	Sektor nicht gefunden Abbrechen, Ignorieren, Wiederholen

ERKLÄRUNG:
Versuchen Sie diesen Fehler mit der Eingabe von **R** zu beheben. Tritt dieser Fehler öfter auf, oder kann er nicht behoben werden, sollten Sie die Daten des Laufwerkes komplett sichern. Danach formatieren Sie das Laufwerk neu, und kopieren Sie mit dem Dienstprogramm RESTORE Ihre Daten zurück.

Seek error reading/writing drive x	Suchfehler beim Lesen/Schreiben des Laufwerkes x

ERKLÄRUNG:
Das F.A.T. des bezeichneten Laufwerkes ist beschädigt, oder der Laufwerkkontroller ist defekt.

Sharing violation reading/writing drive x	Datei gesperrt beim Lesen/Schreiben des Laufwerkes x

ERKLÄRUNG:
Ihr Programm versucht auf eine Datei eines Programmes zuzugreifen, die von einem anderen Programm bereits gesperrt wurde. Warten Sie einen Moment (bis die Datei wieder freigegeben ist), und wiederholen Sie dann mit **R**(etry), oder brechen Sie mit A(bort) ab.

Syntax error	Schreibfehler

ERKLÄRUNG:
Die Schreibweise des eingegebenen Befehls ist falsch.

Unable to create directory	Verzeichnis kann nicht angelegt werden

ERKLÄRUNG:
Der Name des neuen Verzeichnisses enthält ungültige Zeichen oder ist bereits für ein anderes Verzeichnis vergeben.

Fehlermeldungen

Write fault writing drive x	Schreibfehler beim Schreiben des Laufwerkes x

ERKLÄRUNG:
Die Diskette des bezeichneten Laufwerkes ist beschädigt. Prüfen Sie mit CHKDSK, und nehmen Sie eventuell eine andere Diskette.

Write protect writing drive x	Diskette schreibgeschützt im Laufwerk x

ERKLÄRUNG:
Die Diskette ist mit einem Schreibschutz versehen und kann daher nur noch gelesen werden. Erst wenn Sie den Schutz entfernt haben (s. a. Kapitel 1), können Sie die Daten der Diskette ändern oder neu speichern.

Zusammenfassung

Das Betriebssystem MS-DOS kennt zwei verschiedene Befehlsarten: Interne Befehle und Externe Befehle. Interne Befehle sind in der Datei COMMAND.COM enthalten und können daher immer aufgerufen werden. Externe Befehle sind eigene kleine Programme und werden bei jedem Aufruf erst in den Arbeitsspeicher geladen.

Eine Befehlszeile besteht aus den Teilen:

D:	Name des Laufwerkes
Weg	Name des Verzeichnisses
Funktion	Befehl oder Batchdatei
Variable	Bestimmt die Arbeitsweise der Funktion genauer
RETURN	Beendet die Eingabe in die Befehlszeile

Die Eingabe der Befehle muß nach bestimmten Regeln erfolgen. Andernfalls werden die Befehle nicht ausgeführt, es erscheint eine entsprechende Fehlermeldung.

Falsch eingegebene Befehle können Sie mit Hilfe des Tastatureingabespeichers und der Funktionstasten leicht korrigieren.

Befehle können in einer eigenen Datei gespeichert werden. Sie werden dann nach Aufruf dieser Datei automatisch verarbeitet.

Durch Eingabe von speziellen Tastenbefehlen kann die Abarbeitung der Befehle beeinflußt werden.

Die Tastenkombination **CTRL-BREAK** bricht die Bearbeitung eines Befehls sofort ab.

Kapitel 6
Systemstart

Übersicht

Wie bereits in Kapitel 2 erklärt, wird nach dem Einschalten das Betriebssystem von einem externen Speicher (Diskette oder Winchester) in den Arbeitsspeicher kopiert. Den Teil des Arbeitsspeichers mit dem Betriebssystem nennt man Systemspeicher. Der Kopiervorgang wird Systemstart oder auch *Bootup* genannt. Ist der Vorgang erfolgreich abgeschlossen, können Sie Befehle eingeben.

Ihr Zenith PC kann auf zwei verschiedene Arten *gestartet* (gebootet) werden:

- automatisch
- durch einen eigenen Bootbefehl

Bei dem Zenith Z-200 können Sie dies mit dem Monitorprogramm SETUP einstellen. Bei allen anderen Zenith PC können Sie dies mit einem *Schalter* auf der Hauptplatine in der Zentraleinheit einstellen. Weiterhin können Sie durch die Schalterstellung bestimmen, ob von einer *Diskette* oder einer eingebauten *Winchester* gebootet werden soll. Mit Hilfe eines Winchesterdienstprogrammes können Sie einen Teil des Winchesterlaufwerkes als Startlaufwerk festlegen.

Einführung

Zum besseren Verständnis des folgenden Kapitels ist es erforderlich, die Begriffe *Software, Hardware* und *Firmware* näher zu erklären.

Software

Mit diesem Begriff beschreibt man sämtliche *Programme* des Computers. Das können sein: das Betriebssystem, ein Anwenderprogramm oder auch Programme, mit denen andere Programme erzeugt und verändert werden. Mit *Software* weisen Sie den Computer an, bestimmte Funktionen auszuführen.

Hardware

Mit diesem Begriff beschreibt man sämtliche *Geräte*, aus denen sich das Computersystem zusammensetzt. Dies ist die Zentraleinheit, der Bildschirm, die Tastatur oder ein Drucker. Die *Hardware* wird von der *Software* gesteuert.

Firmware

Mit diesem Begriff werden *Programme* (Software) beschrieben, die *firmenspezifisch* in einem eigenen Chip gespeichert sind. Dieser Chip kann nur gelesen werden. Er wird deshalb als Nur-Lese-Speicher (ROM = **R**ead **O**nly **M**emory) bezeichnet.

Startvorgang

In Ihrem Zenith PC ist ein ROM vorhanden mit dem Programm *MFM-MONITOR*. Dieses ist in verschiedene Routinen gegliedert, die Informationen zwischen den einzelnen Teilen Ihres Computers transportieren. Diese Teile werden Ein-/Ausgaberoutinen genannt. Im Monitorprogramm ist auch ein Teilprogramm enthalten, das nach dem Einschalten auf die Platte zugreift, um nach dem Betriebssystem zu suchen.

Einführung

Jede von MS-DOS formatierte Platte enthält im ersten Sektor der der ersten Seite auf der ersten Spur (Seite 0, Spur 0, Sektor 1) ein MS-DOS-eigenes *Ladeprogramm*. Dieses Programm wird BOOT-LOADER genannt. Das Monitorprogramm sucht dieses Ladeprogramm und lädt es in den Arbeitsspeicher. Danach übernimmt es die weitere Kontrolle. Das Ladeprogramm sucht auf der Diskette die Systemdateien an den dafür bestimmten Plätzen und lädt in der folgenden Reihenfolge den Arbeitsspeicher:

- IO.SYS
- MSDOS.SYS
- COMMAND.COM

Danach übernimmt das *Betriebssystem* die Kontrolle. Es überprüft die Hardware (Speichergröße, Peripheriegeräte) und bereitet den Arbeitsspeicher zur Aufnahme von Anwenderprogrammen vor.

Kann dieser Vorgang nicht wie beschrieben ablaufen, so werden entsprechende Fehlermeldungen auf dem Bildschirm ausgegeben. Findet das Monitorprogramm den BOOT-LOADER nicht, erscheint die Meldung:

+++ DISK ERROR: . . .	Diskettenfehler

Findet der BOOT-LOADER nicht das Programm IO.SYS erscheint die Meldung:

No System	Kein System

Startmöglichkeiten

Automatischer Start

Durch das Monitorprogramm SETUP (für Serie Z-200 PC) oder einen Schalter auf der Hauptplatine (für Serie Z-100 PC) wird die *Art* des Startvorganges bestimmt. Nähere Einzelheiten dazu finden Sie in Ihrem entsprechenden Bedienerhandbuch. Ihr Zenith PC ist werkseitig auf *automatischen Startvorgang* vom Laufwerk A: aus eingestellt worden. Es ist daher nicht erforderlich, dies zu verändern.

Sie können die Einstellung leicht überprüfen, ohne das Gerät zu öffnen. Schalten Sie das Gerät ein, und geben Sie keinen Befehl über die Tastatur ein. Erscheint jetzt die Statusmeldung des Monitorprogrammes, so muß der Startvorgang durch eine Befehlseingabe ausgelöst werden. Erscheint die MS-DOS-Statusmeldung oder die Meldung:

DISK ERROR   Diskettenfehler

so wird das System automatisch gestartet.

Sie können einen automatischen Startvorgang abbrechen, wenn Sie sofort nach dem Einschalten des Gerätes die Taste **ESC** oder die Tasten **CTRL-BREAK** mehrmals betätigen.

Start durch Bootbefehl

Haben Sie sämtliche an Ihren Zenith PC angeschlossenen Peripheriegeräte eingeschaltet, schalten Sie die Zentraleinheit ein. Ist Ihr Gerät auf Start durch einen Bootbefehl eingestellt, erscheint nach einigen Sekunden in der linken oberen Ecke Ihres Bildschirmes die Statusmeldung des Monitorprogrammes:

```
MFM-MONITOR, Version 0.8
Memory Size: 512K bytes
Enter "?" for help.
-->
```

Sie können jetzt den Startbefehl eingeben. Dies muß mindestens der Buchstabe **B** (Boot) sein.

Geben Sie hier das Fragezeichen ein, erscheint das Auswahlmenü des Monitorprogrammes. Dabei wird auch die Befehlszeile für den Startbefehl mit aufgeführt. Das Programm wird im Gerätehandbuch näher erklärt.

Der Bootbefehl setzt sich aus folgenden Teilen zusammen:

B [Platte] [Nummer] [:Teil der Winchester]

dabei bedeutet:

B	Beginn des Bootvorganges
Platte	Art des Laufwerkes, von dem aus gebootet werden soll; **F** bootet von *Diskette*, **W** bootet von Winchester.
Nummer	Die Nummer des *Bootlaufwerkes*. Bei vier Diskettenlaufwerken sind es: **0, 1, 2, 3**; bei acht Winchesterlaufwerken: **0, 1, 2, 3, 4, 5, 6, 7**. Wird keine Nummer eingegeben, wird jeweils eine 0 angenommen.

:Winchester-teil	Bezeichnet den Startteil der Winchester von 1 - 4. Der Doppelpunkt muß immer mit eingegeben werden.

Die Eingabe wird mit Betätigen der Taste **RETURN** abgeschlossen.

Variable

Das Monitorprogramm erlaubt Ihnen verschiedene Möglichkeiten der Befehlseingabe. Außer dem Buchstaben **B** für Boot ist es nicht erforderlich, weitere Angaben zu machen. Das Monitorprogramm greift dann auf gespeicherte Werte zurück.

Nach dem Einschalten des Gerätes sind dies die Werte, die durch den Schalter der Hauptplatine oder das Monitorprogramm vorgegeben werden. Wird durch die Tastenkombination **CTRL-ALT-DEL** neu gestartet, so werden alle Werte auf Null zurückgesetzt, und das System verhält sich wie beim Einschalten.

Wird bei eingeschaltetem Gerät mit der Tastenkombination **CTRL--ALT-INS** neu gestartet (*Warmboot*), erscheint die Statusmeldung des Monitorprogrammes. Nach Eingabe eines **B** für Boot, greift das Monitorprogramm dann auf die Werte des zuletzt eingegebenen Startbefehles zurück.

Kurzeingabe

Die kürzeste Form des Bootbefehls ist die Eingabe von:

```
B
```

Je nach Schalterstellung, gespeicherten Werten des Programmes SETUP oder einem eventuell schon eingegebenen Bootbefehl, ergänzt das Monitorprogramm die Befehlszeile automatisch.

Soll von der Diskette gebootet werden, so ist dies nach dem Einschalten des Gerätes

BF0

Soll von der Winchester gebootet werden, so ist dies nach dem Einschalten des Gerätes

BW0:1

Beispiele

Die beiden folgenden Tabellen zeigen eine Übersicht der möglichen Eingaben. Der letzte Befehl bootet jeweils das System, ohne auf vorgegebene Werte zurückzugreifen.

Tabelle 6.1 Bootbefehle für Diskettenlaufwerke

LAUFWERKNUMMER	BEFEHLSEINGABE
0	B **RETURN** BF **RETURN** B0 **RETURN** BF0 **RETURN**
1	B1 **RETURN** BF1 **RETURN**
2	B2 **RETURN** BF2 **RETURN**
3	B3 **RETURN** BF3 **RETURN**

Startmöglichkeiten

Tabelle 6.2 Bootbefehle für Winchesterlaufwerke

LAUFWERKNUMMER	TEIL	BEFEHL
0	1	BW **RETURN** BW0 **RETURN** BW0:1 **RETURN**
	2	BW:2 **RETURN** BW0:2 **RETURN**
	3	BW:3 **RETURN** BW0:3 **RETURN**
	4	BW:4 **RETURN** BW0:4 **RETURN**
1	1	BW1 **RETURN** BW1:1 **RETURN**
	2	BW1:2 **RETURN**
	3	BW1:3 **RETURN**
	4	BW1:4 **RETURN**

Für weitere Laufwerke gelten die Eingaben entsprechend.

Damit das System vom ersten Winchesterlaufwerk aus gebootet werden kann, ist es unbedingt erforderlich, mit dem Dienstprogramm PART einen Teil des Laufwerkes als Startlaufwerk festzulegen.

Ergebnisse

Nach Eingabe des Bootbefehles wird das Betriebssystem MS-DOS in den Arbeitsspeicher geladen. Ist dieser Vorgang erfolgreich abgeschlossen, meldet sich das Betriebssystem und erwartet Ihre Eingaben. Ist keine Datei AUTOEXEC.BAT vorhanden, mit der eventuell ein Anwenderprogramm geladen wird, meldet sich das System mit der *Statusmeldung*:

```
MS-DOS Version 3.10
Copyright 1981,82,83,85 Microsoft Corp.

Current date is Mon 01-01-1980
Enter new date:_
```

Geben Sie das aktuelle Tagesdatum, sowie nach einer weiteren Aufforderung die aktuelle Uhrzeit ein. Danach erscheint das Prompt mit Angabe des Startlaufwerkes. Die Tabelle 6.3 gibt eine Übersicht der möglichen Laufwerkbezeichnungen.

Tabelle 6.3 Mögliche Laufwerkbezeichnungen

Anzahl der Diskettenlaufwerke	Diskettenname	Winchestername
1 oder 2	A (und B)	C, D, E und F
3	A, B und C	D, E, F und G
4	A, B, C und D	E, F, G und H

Laufwerkbezeichnungen bestehen aus den Buchstaben A bis H mit nachfolgendem Doppelpunkt (:).

Die Buchstaben A und B bezeichnen immer ein Diskettenlaufwerk, die Buchstaben C und D nur dann, wenn weitere Diskettenlaufwerke angeschlossen sind, sonst Teile des Winchesterlaufwerkes. Die Buchstaben E bis H bezeichnen immer Teile eines Winchesterlaufwerkes.

Ergebnisse

Ist Ihr Zenith PC mit einem Winchesterlaufwerk ausgerüstet, müssen Sie eine Besonderheit beachten.

Haben Sie das Betriebssystem auf automatische Bezeichnung der Winchesterteilbereiche gestellt, wird immer nur ein Teilbereich des Laufwerkes bezeichnet. MS-DOS reserviert keine weiteren Buchstaben für die anderen Bereiche. Mit dem Programm ASGNPART können Sie aber beliebig oft die Teilbereiche wechseln. Dabei gilt, daß immer nur auf einen Teilbereich der Winchester gleichzeitig zugegriffen werden kann.

Dies ist für verschiedene neuere Programme erforderlich. Nach ihrem Aufruf erfragen diese Programme von MS-DOS die Zahl der vorhandenen Laufwerke ab und überprüfen sie anschließend sofort.

Haben Sie mit dem Programm CONFIGUR das Betriebssystem auf manuelle Bezeichnung gestellt, können Sie mit maximal vier Winchesterteilbereichen gleichzeitig arbeiten. Der erste Teilbereich wird während des Systemstartes bereits bezeichnet, alle anderen müssen Sie mit dem Programm ASGNPART bezeichnen. Danach können Sie auf alle vier Bereiche gleichzeitig zugreifen.

Fehlermeldungen

Kann das Betriebssystem nicht mit den beschriebenen Befehlen gestartet werden, erscheint eine Fehlermeldung. Je nach Fehlerquelle wird die Meldung vom Monitorprogramm oder von einem MS-DOS-Programm erzeugt.

Fehlermeldungen des Monitorprogrammes

Entdeckt das Monitorprogramm einen Fehler in der Eingabezeile, wird dieses durch die Meldung "^Invalid Command!" angezeigt. Das Zeichen "^" steht dabei unter dem erkannten Fehler. Geben Sie die Zeile dann neu ein.

Erscheint eine der folgenden Fehlermeldungen, müssen Sie den Bootvorgang neu beginnen. Dabei gibt es drei mögliche Tastenkombinationen:

- **CTRL-BREAK**, geht zur Eingabe zurück
- **CTRL-ALT-INS**, startet neu und speichert bisher eingegebene Werte (*Warmboot*)
- **CTRL-ALT-DEL**, startet völlig neu (*Kaltboot*)

```
+++ DISK ERROR:  Drive not ready!  +++
+++ DISK ERROR:  Seek failure!  +++
+++ DISK ERROR:  Cannot reset drive! +++
+++ DISK ERROR:  Invalid data read! +++
+++ DISK ERROR:  Data corrected! +++
```

Diese Meldungen sehen Sie, wenn der Computer versucht, das Betriebssystem von der Diskette zu laden, ohne daß sich eine Diskette im Laufwerk befindet. Prüfen Sie, ob eine Diskette ordnungsgemäß eingelegt wurde und das Laufwerk richtig verschlossen ist.

```
+++ DISK ERROR:  Disk not bootable! +++
```

Die Diskette des Startlaufwerkes kann fehlerfrei gelesen werden. Es ist aber nicht das Programm BOOT-LOADER auf dem ersten Sektor der Diskette gespeichert.

Fehlermeldungen

+++ DISK ERROR: Must run SETUP to boot from Winchester! +++

Diese Meldung erscheint, wenn Sie versuchen, von einem Winchesterlaufwerk zu starten, von dem keine Werte gespeichert sind. Rufen Sie erst das Monitorprogramm SETUP auf, und wählen Sie den entsprechenden Laufwerktyp aus.

+++ DISK ERROR: Bad disk controller! ++
+++ DISK ERROR: DMA overrun error! +++

Normalerweise melden diese Angaben einen Fehler des Diskettenkontrollers. Es kann sich aber auch um fehlerhafte Zusatzkarten handeln.

+++ DISK ERROR: Sector not found! +++
+++ DISK ERROR: CRC error! +++
+++ DISK ERROR: Invalid address mark detected! +++

Diese Fehler kommen vor, wenn das Betriebssystem von der Diskette geladen wird. Die Gründe können eine Diskette ohne Betriebssystem oder ein defektes Laufwerk sein. Meistens lassen sich diese Fehler durch Verwendung einer anderen Diskette beheben.

Fehlermeldungen der MS-DOS-Programme

Bad or missing Command Interpreter	Datei COMMAND.COM fehlt oder ist falsch

ERKLÄRUNG:
Der Befehlsinterpreter des Betriebssystemes fehlt auf dem Startlaufwerk. Es kann nicht weiter gebootet werden. Das kann die Datei COMMAND.COM sein, oder aber ein anderer Interpreter, der durch die Datei CONFIG.SYS aufgerufen wird. Kopieren Sie die richtige Datei auf das Startlaufwerk, und booten Sie erneut.

I/O error	Ein-/Ausgabefehler

ERKLÄRUNG:
Es handelt sich um einen Lesefehler der Diskette oder Winchester. Booten Sie von einem anderen Laufwerk, und überprüfen Sie mit CHKDSK die Diskette oder DETECT die Winchester auf eventuelle Beschädigungen.

No bootable partitions	Keine bootfähigen Teile

ERKLÄRUNG:
In der Laufwerktabelle der Winchester wurde kein Startlaufwerk eingetragen. Starten Sie mit einem kompletten Bootbefehl, und definieren Sie anschließend ein Startlaufwerk.

No System	Kein System

ERKLÄRUNG:
Es wird versucht, das System von einer Platte zu starten, die zwar den BOOT-LOADER enthält, aber keine Systemdateien. Der BOOT-LOADER sucht die Systemdateien an einem genau bestimmten Platz. Findet er sie dort nicht, erscheint diese Meldung.

Not a bootable partition	Kein bootfähiger Teil

ERKLÄRUNG:
Das angegebene Laufwerk der Winchester enthält keinen BOOT--LOADER. Formatieren Sie dieses Laufwerk neu mit dem Befehl FORMAT /S.

Partition not formatted	Winchesterteil nicht formatiert

ERKLÄRUNG:
Das Bootlaufwerk der Winchester ist nicht formatiert. Das kann vorkommen, wenn mit dem Programm PART die Winchester neu aufgeteilt wurde.

Fehlermeldungen

Unable to read boot code from partition — Das Boot-Byte kann vom Laufwerk nicht gelesen werden

ERKLÄRUNG:
Die nötigen Informationen zum Booten des Systems sind nicht auf dem Teil der Winchester, oder der Sektor ist fehlerhaft. Booten Sie von einem anderen Laufwerk, und formatieren Sie den defekten Teil neu. Tritt der Fehler wieder auf, muß die gesamte Winchester neu initialisiert werden (Dateisicherung nicht vergessen!). Tritt der Fehler immer noch auf, wenden Sie sich an Ihren Händler.

Unrecognized command in CONFIG.SYS — Befehl in der Datei CONFIG.SYS nicht erkannt

ERKLÄRUNG:
In der Datei CONFIG.SYS wurde ein dort nicht erlaubter Befehl gelesen. Nähere Informationen dazu im Kapitel 9.

Zusammenfassung

Das firmeneigene Monitorprogramm MFM kann das Betriebssystem von jedem angeschlossenen Laufwerk booten. Es sucht auf dem angegebenen Laufwerk nach dem MS-DOS BOOT-LOADER. Dieser wird in den Arbeitsspeicher geladen und kopiert dann das restliche Betriebssystem von der Platte in den Speicher.

Die verschiedenen Möglichkeiten, Ihren Zenith PC zu booten, können Sie mit dem Schalter SW-102 auf der Hauptplatine (Serie Z-100 PC) oder dem Monitorprogramm SETUP (Serie Z-200 PC) vorgeben. Sie können automatisch von der Diskette booten oder von der Winchester. Ebenso können Sie durch einen eigenen Bootbefehl starten.

Der Bootbefehl braucht nicht komplett eingegeben zu werden. Die nicht eingegebenen Teile werden automatisch vom Monitorprogramm ergänzt. Führen Sie einen Warmboot durch, werden die Werte vom letzten Bootbefehl genommen.

Nach dem Systemstart wird, falls vorhanden, sofort die Datei AUTOEXEC.BAT abgearbeitet und nicht nach dem Datum und der Zeit gefragt.

Meldet sich das System mit dem Prompt, wird das Bootlaufwerk angezeigt. Dies wird mit einem Buchstaben von A bis H bezeichnet. Die ersten Buchstaben, maximal bis D, bezeichnen ein Diskettenlaufwerk, die folgenden ein Winchesterlaufwerk oder einen Teil davon.

Kapitel 7

Disketteninhaltsverzeichnis

Übersicht

In diesem Kapitel wird die *Struktur* des Inhaltsverzeichnisses erklärt. Diese Struktur gleicht einem *Baum*, dessen Zweige zu den gespeicherten Dateien führen. Es handelt sich hierbei um ein geordnetes Konzept, das mit der MS-DOS Version 2.0 eingeführt wurde. Es wird für alle weiteren Versionen gültig bleiben. Das System wird auch als *hierarchische Struktur* des Inhaltsverzeichnisses bezeichnet. Der Zugriff auf einzelne Dateien ist daher nicht mehr so einfach wie bisher möglich. Es gibt mehrere verschiedene Befehle, die sich speziell auf das Inhaltsverzeichnis beziehen. Diese Befehle ermöglichen den Zugriff auf jede Datei an jeder beliebigen Stelle.

Die Verwaltung der Verzeichnisse beruht auf einem Konzept, das sich mit einem Baum vergleichen läßt. Genau wie ein Baum, hat auch dieses System eine Wurzel.

Von dieser Wurzel aus wächst das System. Da dies nicht von alleine geschieht, gibt es dafür spezielle Befehle.

Vom ersten *Inhaltsverzeichnis* einer jeden Platte aus, können Sie *weitere Verzeichnisse* anlegen. Das sind genauso Dateien wie Ihre Anwenderprogramme oder auch die Dienstprogramme. Da diese Dateien mit einem besonderen Kennzeichen versehen sind, werden sie von MS-DOS als Inhaltsverzeichnisse erkannt. Diese Verzeichnisse sind *Subverzeichnisse.*

Aufbau des Inhaltsverzeichnisses

Alle Versionen von MS-DOS ab der Nummer 2.0 haben ein hierarchisches Inhaltsverzeichnis, in dem die Informationen über Dateien gespeichert werden. Dies bedeutet, daß ein System mit *mehreren Ebenen* vorhanden ist. Die oberste Ebene hat die höchste Priorität, die unterste die niedrigste.

Das System läßt sich mit einem Baum vergleichen, der auf dem Kopf steht. Die Wurzel ist oben, die Zweige wachsen nach unten.

Die Spitze des Systems ist die Wurzel. Das ist hier das erste Verzeichnis, das auf der Platte angelegt wird. Dies geschieht automatisch mit dem Befehl FORMAT. Dieses erste Verzeichnis wird *Rootdirectory* genannt. In diesem Verzeichnis können sowohl Dateien als auch weitere Verzeichnisse gespeichert werden.

Weitere Verzeichnisse, die Sie mit einem speziellen Befehl anlegen, werden *Subdirectory* genannt.

In diesem strukturierten System von Inhaltsverzeichnissen (Directories) können Sie sich in jeder beliebigen Richtung bewegen. Sie können an der Spitze beginnen und bis zu einem Ende gelangen, z. B. wenn Sie eine bestimmte Datei suchen. Haben Sie die Datei nicht gefunden, können Sie zurück und weiter bis zur nächsten Spitze gehen. Ebenso können Sie in der Mitte anfangen und zu jedem beliebigen Ende gehen.

Sämtliche Dateien, die unter MS-DOS geschaffen werden, erhalten einen *Namen*. Dieser Name wird zusammen mit weiteren Informationen über diese Datei in einem Directory gespeichert. Von den weiteren Informationen können Sie die Dateigröße, das Datum und die Zeit der Dateianlage sehen. Dafür geben Sie den Befehl DIR ein. Er zeigt Ihnen sämtliche Dateien an, die in diesem Directory gespeichert sind.

Sie können im *Rootdirectory* einer Diskette mit dem Format DS/DD bis zu 112 Dateien speichern. Eine Diskette des Formates SS/DD kann bis zu 64 Dateien im Rootdirectory speichern. Das Rootdirectory eines Winchesterlaufwerkes kann bis zu 1024 Dateien speichern, je nach Größe des einzelnen Teilbereiches. Die von Ihnen angelegten Subdirectories können jede beliebige Anzahl von Einträgen speichern, begrenzt nur noch durch die Kapazität der Platte.

Aufbau des Inhaltsverzeichnisses

Damit erleichtert Ihnen MS-DOS 2.0 die Verwaltung vieler Dateien. Sie haben die Möglichkeit, Dateien in verschiedenen Subdirectories zu speichern. Diese Subdirectories sind leichter zu überschauen, sie können bei Bedarf so groß wie nötig werden.

Legen Sie ein neues *Subdirectory* an, geben sie ihm einen Namen. Damit wird dieses Subdirectory wie eine Datei behandelt. Geben Sie den Befehl DIR ein, werden nicht nur alle Dateien des Verzeichnisses angezeigt, sondern auch alle Subdirectories, die von diesem Directory aus angelegt wurden. Das Bild 7.1 zeigt ein typisches Inhaltsverzeichnis mit verschiedenen Subdirectories.

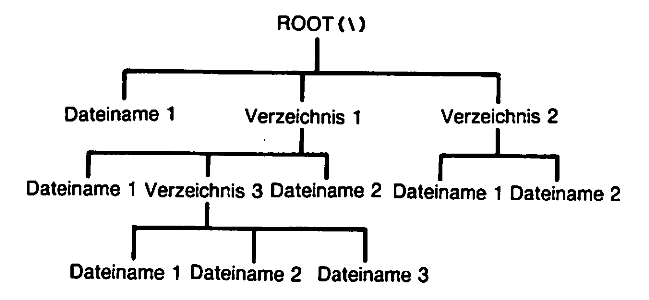


Bild 7.1
Strukturiertes Inhaltsverzeichnis

Das Rootdirectory beinhaltet eine Datei und zwei Subdirectories. Geben Sie den Befehl DIR ein, werden nur diese drei Einträge angezeigt. Das Subdirectory Verzeichnis 1 enthält drei Einträge, zwei Dateien und ein weiteres Subdirectory. Das zweite Verzeichnis des Rootdirectory enthält nur zwei Dateinamen.

In diesem Beispiel haben verschiedene Dateien denselben Namen. Dies ist möglich, da sie sich in verschiedenen Subdirectories befinden. Nur die Dateien in einem Verzeichnis müssen unterschiedliche Namen haben, um sie auch als unterschiedliche Dateien zu erkennen.

Disketteninhaltsverzeichnis

Aufbau des Inhaltsverzeichnisses

Legen Sie das Bild Nr. 7.1 auf den Kopf, werden Sie feststellen, daß es einem *Baum* gleicht. Das Rootdirectory ist die Wurzel oder das Fundament, auf dem alles aufbaut. Die Subdirectories sind die *Zweige*, die von der Wurzel wachsen. Die einzelnen Dateien sind die *Blätter* der Zweige.

Sowie es ein Arbeitslaufwerk gibt, gibt es auch ein momentanes *Arbeitsverzeichnis*. In diesem wird von MS-DOS eine neue Datei angelegt oder auch eine Datei gesucht. Dieses Verzeichnis können Sie jederzeit mit dem Befehl **CHDIR** (**CH**ange **DIR**ectory) wechseln. Damit wechseln Sie in einen anderen Bereich des Baumes.

Legen Sie ein neues Verzeichnis an, werden von MS-DOS sofort in diesem Verzeichnis zwei Dateien eingetragen. Das geschieht automatisch. Diese Einträge können von Ihnen auch nicht gelöscht werden. Es sind *Arbeitseinträge* für MS-DOS, die das momentane Verzeichnis speichern, sowie das Verzeichnis, von dem aus es angelegt wurde. Der Eintrag „ . “ bezeichnet das *momentane Subdirectory*. Der Eintrag „ .. “ bezeichnet das *vorherige Directory*. Damit speichert MS-DOS, in welcher Beziehung die einzelnen Directories zueinander stehen. Geben Sie einen einzelnen Punkt als Verzeichnisnamen ein, bezeichnen Sie damit das Arbeitsverzeichnis. Geben Sie zwei Punkte ein, bezeichnen Sie damit immer das vorherige Verzeichnis. Sind Sie beispielsweise am Ende eines Zweiges und geben mehrmals den Befehl **CHDIR ..** ein, gehen Sie schrittweise zum Rootdirectory zurück.

Wollen Sie auf eine Datei zugreifen, die nicht im momentanen Verzeichnis enthalten ist, müssen Sie nicht unbedingt schrittweise von Subdirectory zu Subdirectory gehen. Sie können in der Befehlszeile die Namen der Subdiretories aufzählen, die zwischen dem momentanen Directory und dem Directory der Datei liegen. Diese Aufzählung wird *Weg* (Path) genannt. Diesen Weg geht MS-DOS, um zur angegebenen Datei zu gelangen. Die einzelnen Verzeichnisse werden bei der Aufzählung jeweils durch einen umgekehrten Schrägstrich (\) getrennt. An das Ende der Aufzählung wird der Dateiname angehängt. Ein Beispiel:

[\] [Verzeichnis] [\Verzeichnis] [\Verzeichnis...] [\Dateiname]

Dabei darf der Name des Weges nicht mehr als 64 Zeichen enthalten. Wird der Weg in Verbindung mit einem Befehl gebraucht, der Verzeichnisse ändert, darf am Ende kein Dateiname eingegeben werden. Gehen Sie auf diesem Weg von der Spitze zur Wurzel zurück, müssen Sie an Stelle des Namens zwei Punkte eingeben.

Beginnt der Weg im momentanen Arbeitsverzeichnis, ist es ein *relativer Weg*. Der *absolute Weg* beginnt im Rootdirectory. Dies wird MS-DOS durch die Eingabe eines umgekehrten Schrägstriches zu Beginn mitgeteilt. Ein Beispiel:

```
\Verzeichnis1\Verzeichnis3\Dateiname1
```

Viele Anwenderprogramme wurden geschrieben, als es MS-DOS 2 mit dem hierarchischen Inhaltsverzeichnis noch nicht gab. Sie arbeiten unter MS-DOS 2 einwandfrei, mit einer Ausnahme:

Sie sind nicht in der Lage auf Dateien zuzugreifen, die sich in einem anderen Subdirectory befinden. Deshalb müssen sämtliche Dateien, die diese Programme benötigen, im selben Subdirectory gespeichert sein. Das gilt auch für Programme, die sich über mehrere Dateien erstrecken. Der Weg kann von MS-DOS nur für Dateien mit dem Zusatz .COM, .EXE oder .BAT benutzt werden. Er kann nicht für Dateien mit anderen Zusätzen benutzt werden.

Befehle des Inhaltsverzeichnisses

Da die Struktur des Inhaltsverzeichnisses ab der Version 2 dynamisch geworden ist, gibt es besondere Befehle, mit denen Sie diese Struktur verändern können.

Dies sind die folgenden vier Befehle:

- MKDIR legt ein neues Verzeichnis an
- CHDIR ändert das momentane Verzeichnis
- RMDIR löscht ein Verzeichnis
- PATH bezeichnet einen Weg, den MS-DOS durch die Verzeichnisse bis zu einem Programm mit dem Zusatz .COM, .EXE oder .BAT geht.

Befehle des Inhaltsverzeichnisses

Außer diesen speziellen Befehlen sind die folgenden Befehle für das Arbeiten mit Inhaltsverzeichnissen von großer Bedeutung:

- COPY Mit diesem Befehl können Dateien von einem Verzeichnis zum anderen kopiert werden. Dabei wird der Befehl so benutzt, wie beim Kopieren zwischen Laufwerken. Das Ziellaufwerk ist in diesem Fall das Zielverzeichnis.
- JOIN Mit diesem externen Befehl können Sie eine Laufwerkbezeichnung durch einen Verzeichnisnamen ersetzen. Damit können Sie zusätzlich alle Dateien des Laufwerkes in allen Unterverzeichnissen über diesen neuen Verzeichnisnamen erreichen. Die vorherigen Bezeichnungen bleiben weiterhin gültig.
- SUBST Mit diesem externen Befehl können Sie einen Verzeichnisnamen durch eine Laufwerkbezeichnung ersetzen. Damit können Sie zusätzlich alle Dateien des Verzeichnisses über diese neue Laufwerkbezeichnung erreichen. Der vorherige Verzeichnisname bleibt weiterhin gültig.

Weg und externe Befehle

Externe Befehle sind eigene Programme, die beim Aufruf erst in den Arbeitsspeicher geladen werden müssen.

Zur besseren Übersicht sollten Sie Ihre Programme auf mehrere verschiedene Verzeichnisse verteilen. Es bietet sich dabei an, sämtliche Systemprogramme in einem Verzeichnis zu speichern und dieses Verzeichnis dann z. B. SYSTEM zu nennen. Geben Sie jetzt, während Ihrer Arbeit im momentanen Verzeichnis, den Namen eines externen Befehles ein, so sucht MS-DOS dieses Verzeichnis nach dem Programm ab. Es findet das Programm nicht und gibt eine entsprechende Fehlermeldung aus. Es muß jetzt ein *Weg* bezeichnet werden, auf dem MS-DOS vom Arbeitsverzeichnis zum Verzeichnis SYSTEM gelangt.

Sind Sie momentan im Verzeichnis \SYSTEM\PROG und sämtliche Systemdateien sind im Verzeichnis \SYSTEM, so müssen Sie MS-DOS anweisen, im Verzeichnis SYSTEM zu suchen, falls das Dienstprogramm nicht im derzeitigen Verzeichnis ist. Die Eingabe lautet (s. a. Kapitel 11):

```
PATH = \SYSTEM
```

Dieser Weg muß nur einmal eingegeben werden. Soll MS-DOS andere Programme in anderen Verzeichnissen suchen, so kann der Weg jederzeit geändert werden. Durch Eingabe von PATH ohne weitere Zusätze können Sie sich den derzeitigen Weg anzeigen lassen. Ebenso ist es möglich, mehr als einen Weg zugleich festzulegen.

Dieser Weg wird nur für Programme mit den Namenszusätzen .COM, .EXE und .BAT abgesucht. Alle anderen Programme müssen sich, soll auf sie zugegriffen werden, im Arbeitsverzeichnis befinden. Das gilt vor allem für Anwenderprogramme, die in mehrere Teile gesplittet sind, oder die bei Bedarf zusätzliche Programme laden können.

Weg und interne Befehle

Interne Befehle werden durch den Befehlsinterpreter sofort ausgeführt. Einige dieser internen Befehle können Sie ebenfalls mit einem Weg zusammen benutzen. Es handelt sich dabei um die folgenden Befehle:

- COPY
- DEL
- DIR
- RENAME
- TYPE

Nähere Einzelheiten dazu finden Sie im Kapitel 11.

Zugriff auf ein Inhaltsverzeichnis

Die folgenden Abschnitte erklären ausführlich, wie Sie die Struktur der Verzeichnisse ändern können, wie Sie Verzeichnisse anlegen, ändern, oder löschen.

Anzeigen des Verzeichnisses

Jedes Verzeichnis, in dem Sie sich gerade befinden, hat einen Namen, mit einer Ausnahme: das erste Verzeichnis (Rootdirectory), es wird nur mit einem Strich (\) bezeichnet. Möchten Sie wissen, in welchem Directory Sie sich gerade befinden, geben Sie den Befehl

CHDIR oder CD

ohne weitere Zusätze ein. Es erscheint dann der komplette Name des Verzeichnisses mit der Angabe des Laufwerkes, z. B.:

C:\SYSTEM\PROG\SPEICHER

Möchten Sie wissen, welche Dateien Sie in einem Verzeichnis gespeichert haben, geben Sie den Befehl DIR ein, und es könnte folgendes erscheinen:

```
Volume in drive C is ZENITH1
Directory of C:\TEXT

.                  <DIR>      8-05-84      8:54p
..                 <DIR>      8-05-84      8:54p
SPEICHER           <DIR>      8-05-84      8:56p
WS          COM     20864     7-26-84      5:04p
WSOVLY1     OVR     41216     1-01-80     12:00a
WSMSGS      OVR     25344     1-01-80     12:00a
            6 File(s)   1056768 bytes free
```

Das ist das Verzeichnis des Verzeichnisses TEXT. Es befindet sich auf dem Laufwerk C. Dieses Laufwerk wurde ZENITH1 genannt. Zuerst werden die beiden Arbeitseinträge von MS-DOS für das derzeitige Verzeichnis und das vorhergehende Verzeichnis angezeigt. Dann kommt ein weiteres Subdirectory, das von hier aus angelegt wurde, mit dem Namen SPEICHER. Die hier gespeicherten Dateien sind das Textverarbeitungsprogramm WORDSTAR.

Ändern des Verzeichnisses

Sie können sehr leicht das momentane Arbeitsverzeichnis wechseln. Geben Sie dafür den Befehl

CHDIR (**CH**ange **DIR**ectory)

oder die Kurzform

CD

ein. Sie *wechseln* dann von einem Subdirectory zu einem anderen Subdirectory. Es ist auch möglich, über mehrere Subdirectories zu wechseln. Das gilt sowohl für das Vorwärtsgehen (von Wurzel zu Zweig), als auch für das Rückwärtsgehen (von Zweig zu Wurzel). Dabei ist folgendes zu beachten:

Gehen Sie *vorwärts*, müssen Sie den Namen des folgenden Subdirectory eingeben. Gehen Sie *rückwärts*, geben Sie für jedes Subdirectory nur zwei Punkte (..) ein. *Wechseln* Sie von einem Zweig über das Rootdirectory zu einem anderen Zweig, so ist eine gemischte Eingabe erforderlich. Dabei ist jedes Directory mit einem umgekehrten Schrägstrich zu trennen. Ein Beispiel:

CD SPEICHER

wechselt vom derzeitigen Verzeichnis zum folgenden mit dem Namen SPEICHER.

CD ..

wechselt vom derzeitigen Verzeichnis zum vorherigen Verzeichnis. Sind Sie im Rootdirectory, ist diese Eingabe nicht möglich.

CD ..\..\BASIC\PROG

wechselt erst zwei Verzeichnisse zurück bis zum Rootdirectory. Danach geht es auf einen neuen Zweig (BASIC) und dann in das Subdirectory PROG.

Geben Sie einen falschen Namen ein, erscheint die Fehlermeldung:

Invalid directory — Falsches Verzeichnis

Befehle des Inhaltsverzeichnisses

Anlegen eines Verzeichnisses

Ein neues Verzeichnis kann jederzeit mit dem Befehl

MKDIR (MaKe DIRectory)

oder der Kurzform

MD

angelegt werden. Dabei werden dann von MS-DOS sofort die beiden Arbeitseinträge vorgenommen. Sie können dieses neue Verzeichnis sowohl von einem bestehenden Subdirectory aus anlegen, als auch vom Rootdirectory aus. Ein Beispiel:

MD SPEICHER

erzeugt vom momentanen Subdirectory aus ein neues Subdirectory mit dem Namen SPEICHER.

MD \BASIC\SPEICHER

erzeugt dasselbe Subdirectory vom Rootdirectory aus. Es ist dann ein Subdirectory, das im Verzeichnis BASIC gespeichert wird.

Kann das neue Verzeichnis nicht angelegt werden, weil beispielsweise der Name bereits für eine Datei vorhanden ist, erscheint die Fehlermeldung:

Unable to create directory	Verzeichnis kann nicht angelegt werden

Löschen eines Verzeichnisses

Sie können ein Verzeichnis aus der hierarchischen Struktur nur unter bestimmten Voraussetzungen mit dem Befehl

RMDIR (ReMove DIRectory)

oder der Kurzform

RD

löschen. In diesem Verzeichnis dürfen außer den beiden Arbeitseinträgen keine weiteren Dateien mehr gespeichert sein. Das momentane Verzeichnis kann mit diesem Befehl nicht gelöscht werden. Ebenso kann das Rootdirectory nicht gelöscht werden. Sind diese Voraussetzungen erfüllt, kann das Verzeichnis gelöscht werden. Ein Beispiel:

RD Speicher

löscht das im momentanen Verzeichnis gespeicherte Subdirectory SPEICHER.

RD \BASIC\SPEICHER

löscht vom Rootdirectory aus das Subdirectory SPEICHER.

Sind in einem zu löschenden Verzeichnis noch Dateien enthalten, ist es nicht möglich, dieses Verzeichnis zu löschen. Sie können jedoch mit einem Befehl *sämtliche Dateien* (außer den beiden von MS-DOS angelegten) löschen. Geben Sie dazu ein:

DEL SPEICHER

oder

DEL SPEICHER\ *.*

Beide Eingaben sind identisch. Da jetzt alles gelöscht werden soll, erscheint zur Sicherheit die Frage:

Are you sure (Y/N)?	Sind Sie sicher (J/N)?

und erst nach der Eingabe von **Y** für Ja, werden sämtliche Einträge gelöscht. Geben Sie den Befehl falsch ein, erscheint die Fehlermeldung:

Invalid path, not directory, or directory not empty	Falsches Verzeichnis oder Verzeichnis nicht leer

Beispiel

Zum besseren Verständnis werden sämtliche erklärten Befehle noch einmal an einem Modellverzeichnis erklärt. Das Verzeichnis ist nachfolgend abgebildet.

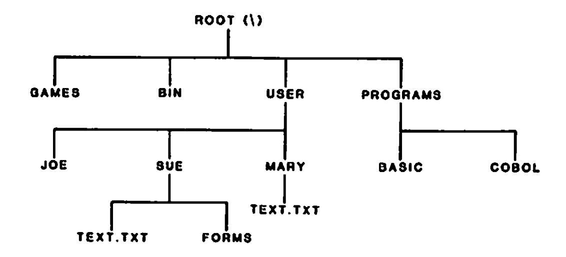


Bild 7.2
Beispiel eines strukturierten MS-DOS-Verzeichnisses

Die erste und zugleich höchste Ebene wird vom Rootdirectory dargestellt. Dies ist das einzige Verzeichnis, das keinen Namen trägt. Es wird in allen Befehlen nur mit einem Strich (\) dargestellt. Vom Rootdirectory gehen vier weitere Subdirectories ab:

- GAMES dort sind Spiele gespeichert
- BIN dort sind alle Dienstprogramme gespeichert
- USER dies enthält weitere Subdirectories für alle Benutzer der Platte
- PROGRAMS hier sind Anwenderprogramme gespeichert

Beispiel

Das Subdirectory USER teilt sich weiter auf in sämtliche Benutzer:

- JOE hier sind noch keine Dateien gespeichert
- SUE mit einem weiteren Subdir und einer Datei TEXT.TXT
- MARY ebenfalls mit einer Datei TEXT.TXT

Die beiden Dateien, die mit TEXT.TXT benannt sind, können verschiedene Inhalte und trotzdem den gleichen Namen haben, da sie in verschiedenen Subdirectories gespeichert sind.

Bei dieser Aufteilung der Platte ist es besonders wichtig, die gesamte Struktur der Platte zu kennen. Sie können sich, unabhängig vom momentanen Verzeichnis, einen Überblick über die Dateien des Subdir FORMS verschaffen, indem Sie eingeben:

```
DIR \USER\SUE\FORMS
```

Durch den ersten Schrägstrich erkennt MS-DOS, daß der Weg durch das System im Rootdir beginnt. Da der zuletzt eingegebene Name ein Subdir benennt, werden jetzt alle Dateien des Subdir aufgelistet. Sollen nur die Daten der Datei TEXT.TXT im Subdir SUE angezeigt werden, geben Sie ein:

```
DIR \USER\SUE\TEXT.TXT
```

Befinden Sie sich im Subdir USER, und Sie wollen sich die Dateien des Subdir FORMS aufzeigen lassen, geben Sie ein:

```
DIR SUE\FORMS
```

Die Eingabe beginnt mit dem Namen eines Subdir. MS-DOS sucht dieses Subdir jetzt im momentanen Verzeichnis und geht dann weiter.

Disketteninhaltsverzeichnis

Beispiel

Wie bereits in einem früheren Abschnitt erklärt, bezeichnen die beiden ersten Einträge eines jeden Subdir jeweils das eigene Subdir (.) und das vorherige Subdir (..), welches sich auf einer höheren Ebene befindet. MS-DOS benötigt diese Einträge, um den Weg zurück zum Rootdir zu finden. Wollen Sie wissen, von welchem Verzeichnis aus Ihr derzeitiges Subdir abgeht, geben Sie ein:

```
DIR ..
```

Wollen Sie die Einträge des eigenen Verzeichnisses sehen, können Sie beide Formen benutzen:

```
DIR
```

oder

```
DIR .
```

Mit den neuen Dienstprogrammen der Version 3 können Sie das gesamte strukturierte Verzeichnis des Beispieles vorübergehend ändern.

Geben Sie den Befehl

```
JOIN C: \BEISPIEL
```

ein, können Sie sämtliche Dateien des Laufwerkes C: (unser Beispiel) auch unter dem Verzeichnisnamen BEISPIEL erreichen. MS-DOS legt dafür auf dem momentanen Laufwerk ein neues SUBDIR an.

Geben Sie den Befehl

```
SUBST E: \USER\SUE
```

ein, können Sie sämtliche Dateien des Verzeichnisses USER\SUE über die Laufwerkbezeichnung E: erreichen.

Zusammenfassung

Die hierarchische Struktur des Inhaltsverzeichnisses ermöglicht eine übersichtliche Speicherung der Dateien. Die Struktur läßt sich mit einem Baum vergleichen, der auf dem Kopf steht. Es können Dateien gleicher Art in einem eigenen Unterverzeichnis (Subdir) abgespeichert werden. Um auf Dienstprogramme zurückgreifen zu können, die nicht im momentanen Verzeichnis gespeichert sind, kann ein Weg eingegeben werden. Es kann von jedem Verzeichnis dieses Baumes in jedes andere gewechselt werden. In der Zusammenfassung noch einmal die wichtigsten Eingabeformen:

\	bezeichnet das Rootdir der Platte
\PROGRAMS	bezeichnet das erste Subdir nach dem Rootdir
\USER\MARY\FORMS\TAX	bezeichnet einen absoluten Weg durch das System, der im Rootdir beginnt. Ist der zuletzt eingegebene Name eine Datei, werden nur die Daten der Datei angezeigt.
USER\SUE	bezeichnet einen relativen Weg durch das System, der im momentanen Subdir beginnt.
.	bezeichnet das momentane Subdir
..	bezeichnet das vorherige Subdir. Wird benutzt, um zum Rootdir zurückzugehen.

Zusammenfassung

Tabelle 7.1 Befehle des Inhaltsverzeichnisses

BEFEHL	ZWECK	EINGABE
MKDIR oder MD	legt ein neues Verzeichnis an	MD Name
CHDIR oder CD	verändert oder zeigt den Namen eines Inhaltsverzeichnisses	CD [Name] CD [..]
COPY	kopiert Dateien	COPY Dateibez [d:] [Weg]
JOIN	ersetzt einen Laufwerknamen durch ein Verzeichnis	JOIN d: [d:] Weg
PATH	benennt mehrere Verzeichnisse in Folge	PATH = Name...
RMDIR oder RD	löscht ein Inhaltsverzeichnis	RD Name
SUBST	ersetzt ein Verzeichnis durch ein Laufwerk	SUBST d: [d:] Weg

Eingabe / Ausgabe

Übersicht

Dieses Kapitel erklärt ausführlich welche Ein-/Ausgabefunktionen das Betriebssystem MS-DOS unterstützt. Es wird gezeigt, welche Funktionen für die einzelnen Geräte vorgegeben sind und wie sich diese Funktionen ändern lassen.

Weiterhin ist der Anschluß eines Druckers an Ihren Zenith PC erklärt und wie sie den Bildschirminhalt auf den Drucker kopieren können.

Ebenso wird beschrieben, wie Sie einen anderen Zeichensatz auf Ihrem Bildschirm ausgeben können.

Ursprung und Ziel

Alle Ein-/Ausgaben werden unter MS-DOS als Informationsaustausch zwischen *Peripheriegeräten* (Devices) behandelt. Peripheriegeräte sind z. B. die Tastatur, der Bildschirm, Laufwerke, Drucker und Modems.

Der Teil des Betriebssystemes, der für die Steuerung der Peripheriegeräte und den reibungslosen Datenaustausch zwischen ihnen zuständig ist, wird *Eingabe-Ausgabe-Lenker* (I/O Handler) genannt. Dies ist hauptsächlich die Datei IO.SYS, aber auch die Datei MSDOS.SYS. Nähere Einzelheiten zum Aufbau der Dateien finden sie im Kapitel 9.

Der I/O Handler benötigt zum ordnungsgemäßen Arbeiten ein Hilfsprogramm, daß auf die speziellen Eigenarten des angeschlossenen Gerätes abgestimmt ist. Dieses Hilfsprogramm wird *Gerätetreiber* (Device-Driver) genannt. Es beinhaltet alle Informationen, die das Betriebssystem über das Peripheriegerät wissen muß. Damit können die Daten des Gerätes übersetzt werden. Anschließend können sie dann gelesen oder geschrieben werden.

MS-DOS unterstützt zwei Arten von Gerätetreibern:

Charaktertreiber Sie ermöglichen eine serielle Ein-/Ausgabe, z. B. für Tastatur, Bildschirm oder serieller Drucker. Die Datei ANSI.SYS ist ein Charaktertreiber (s. a. Kapitel 9).

Blocktreiber Sie ermöglichen den Transport von Datenblökken zwischen den einzelnen Geräten (z. B. Schreiben oder Lesen von Diskette). Die Größe des Blocks wird vom Treiber bestimmt. Dieser Block ist bei den von Zenith Data Systems gelieferten Treibern mit 512 Bytes (1 Sektor) festgelegt. Die Datei VDISK.SYS ist ein Blocktreiber (s. a. Kapitel 9).

Ursprung und Ziel

Charaktertreiber werden durch den Namen des Peripheriegerätes benannt. Sie können nur ein Gerät unterstützen. Diese Namen entsprechen den Namen, die von MS-DOS für Dateinamen akzeptiert werden (maximal acht Buchstaben). Eine Übersicht zeigt die Tabelle 8.1.

Im Gegensatz dazu können *Blocktreiber* mehrere verschiedene Geräte der gleichen Art zur selben Zeit unterstützen. Sie werden nach dem Laufwerkbuchstaben benannt.

Einige der Charaktertreibernamen, die von MS-DOS benutzt werden, sind in der folgenden Tabelle aufgeführt. Diese Namen sind für das Betriebssystem reserviert und dürfen daher nicht von Ihnen für Ihre Dateien benutzt werden.

Tabelle 8.1 Namen der von MS-DOS benutzten Charaktertreiber

NAME	PERIPHERIEGERÄT
AUX	asynchrone Schnittstelle
CLOCK$	Treiber für Echtzeit-Uhr
COM1	erster Seriellausgang
COM2	zweiter Seriellausgang
CON	Console
LPT1	erster Parallelausgang
LPT2	zweiter Parallelausgang
LPT3	dritter Parallelausgang
NUL	nicht vorhandener Ausgang
PRN	Drucker

Der normale MS-DOS *Blocktreiber* unterstützt bis zu acht logische Laufwerke (Diskette oder Winchester), die mit den Buchstaben A bis H bezeichnet werden.

Ursprung und Ziel

Die Namen der *Gerätetreiber* können als Ursprung oder Ziel für eine Ein- oder Ausgabe benutzt werden. Ein Beispiel:

COPY CON Dateiname		Kopiere von Konsole in Datei

Kopiert Daten (Ihre Eingaben) von der Konsole (Tastatur) in eine Datei (z. B. in die Datei AUTOEXEC.BAT).

Der *Gerätetreiber* NUL ist an kein tatsächlich vorhandenes Gerät angeschlossen. Daher werden die Daten, die zu diesem Treiber geschickt werden, nur als Ausgangsinformationen verarbeitet. Sie sind anschließend nicht mehr vorhanden. Ein Beispiel:

LISTPROG FIN.IFO PERS.IFO

ruft ein Anwenderprogramm (*LISTPROG*) auf, daß bei seinem Aufruf die Eingabe zweier Dateinamen (*FIN.IFO* und *PERS.IFO*) für die auszugebenden Daten erfordert. Sie wollen aber diesmal nur die Personendaten abfragen. Sie rufen deshalb das Programm auf:

LISTPROG NUL PRN

Die von der Syntax her geforderten zwei Dateinamen haben Sie eingegeben. Für die Personaldaten (PERS.IFO) haben Sie *PRN* eingegeben, für die Finanzdaten (FIN.IFO) haben Sie *NUL* eingegeben. Es werden jetzt die Personaldaten auf den Drucker ausgegeben, und die Finanzdaten zum Ausgang NUL gesandt.

Standardeingabe und Standardausgabe

Sämtliche MS-DOS-Befehle und -Funktionen lesen eingegebene Daten vom *Standardeingabegerät* und geben Daten zum *Standardausgabegerät* weiter. Der Treiber für diese Standardgeräte ist der Charaktertreiber mit dem Namen CON. Er liest eingegebene Daten vom Standardeingabegerät (Tastatur) und gibt auszugebende Daten zum Standardausgabegerät (Bildschirm). Das sind die vorgegebenen Funktionen im Betriebssystem.

Durch entsprechende Befehle können Sie dies ändern. Sie können die *Standardeingabe* oder die *Standardausgabe* auf verschiedene Arten beeinflussen:

- Datenumleitung
- Verkettung von Befehlen
- Befehlseingabe aus einer Datei

Bei der *Datenumleitung* wird ein anderes Eingabegerät als die Tastatur zum Standardeingabegerät bestimmt. Das kann z. B. ein anderer Computer sein, aber nie ein Drucker. Ebenso wird ein anderes Ausgabegerät als der Bildschirm zum Standardausgabegerät bestimmt. Dies kann ein Drucker oder auch eine Datei sein, aber nie die Tastatur.

Bei der *Verkettung* von Befehlen wird das System angewiesen, die auszugebenden Daten des einen Befehls als einzugebende Daten des anderen Befehles zu nehmen.

Bei der Befehlseingabe aus einer Datei werden alle Befehle des Programmes vorher in einer Textdatei gespeichert. Das Programm arbeitet dann die Datei ab. Eine Eingabe von der Tastatur ist solange nicht möglich.

Datenumleitung

Durch die Eingabe besonderer Zeichen kann das Betriebssystem sehr leicht angewiesen werden, Daten nicht mehr von der Tastatur zu lesen, sondern von einer anderen Ursprungsquelle; Daten nicht mehr zum Bildschirm zu schicken, sondern zu einem anderen Empfangsgerät. Es sind dies die Zeichen:

Datenumleitung

< Ursprung bestimmt, daß der Befehl Daten von diesem Ursprung (Datei oder Peripheriegerät) entgegennimmt

> Ziel bestimmt, daß die Ausgabedaten des Befehls zum angegebenen Ziel (Datei oder Peripheriegerät) gegeben werden

>> Ziel bestimmt, daß die Ausgabedaten des Befehls an das Ziel angehängt werden. Dies ist normalerweise eine Datei.

Eingabe aus einer Datei oder einem Peripheriegerät

Für Befehle, die Ihre Daten vom Standardeingabegerät lesen, kann mit dem Zeichen „ < “ das Standardeingabegerät neu definiert werden. Das kann dann eine Datei oder ein datenaussendendes Peripheriegerät sein. Die *Datenumleitungsfunktion* wird nach dem Befehlsnamen in der Befehlszeile eingegeben. Dies geschieht für Dateien in der Form:

```
Befehl < Dateibez
```

Für entsprechende Peripheriegeräte lautet die Schreibweise:

```
Befehl < Device
```

Ein Beispiel für einen Befehl, der Daten zur Verarbeitung einliest.

```
SORT < ZEICHEN.TXT
```

Der Befehl SORT liest die zu sortierenden Daten von der Datei ZEICHEN.TXT ein. Diese Datei befindet sich im momentanen Inhaltsverzeichnis. Das Standardeingabegerät, von dem aus Daten gelesen werden, ist jetzt nicht mehr die Tastatur, sondern die Datei ZEICHEN.TXT.

Ein weiteres Beispiel für Datenumleitung:

SORT < COM1

Es sollen Daten von einem Modem (COM1) empfangen werden. Sowie die Daten empfangen werden, werden sie sortiert und erst danach auf dem Bildschirm ausgegeben.

Arbeiten Ihre Anwendungsprogramme nicht mit den *Funktionsaufrufen* für die Standardeingabe oder Standardausgabe, so wird die Datenumleitung vom Programm nicht erkannt. Diese Funktion kann dann nicht in Verbindung mit dem Programm benutzt werden.

Ausgabe zu einer Datei oder einem Peripheriegerät

Für Befehle, die Ihre Daten zu einem Standardausgabegerät senden, kann mit dem Zeichen „ > " das Standardausgabegerät neu definiert werden. Dies kann dann eine Datei oder ein datenempfangendes Peripheriegerät sein. Die *Datenumleitungsfunktion* wird nach dem Befehlsnamen in der Befehlszeile eingegeben. Das geschieht für Dateien in der Form:

Befehl > Dateibez

Für entsprechende Peripheriegeräte lautet die Schreibweise:

Befehl > Device

Werden die Daten zu einer Datei geschickt, die noch nicht existiert, so wird diese Datei neu angelegt. Ist die Datei bereits vorhanden, so wird der alte Inhalt überschrieben. Werden die Daten zu einem Peripheriegerät gegeben, muß es mit einem gültigen Namen bezeichnet werden (s. a. Tabelle 8.1).

Datenumleitung

Ein Beispiel für die *Datenausgabe* in eine Datei. Sie wollen die Daten des Inhaltsverzeichnisses speichern und geben sie in eine Datei. Geben Sie ein:

DIR C: > DATEI.DIR

Die Ausgabe, die normalerweise auf dem Bildschirm erfolgen würde, erfolgt jetzt in die Datei DATEI.DIR, auf dem Bildschirm wird nichts angezeigt.

Ebenso ist es möglich, die Daten des Inhaltsverzeichnisses auf den Drucker zu geben. Geben Sie ein:

DIR C: > PRN

Arbeiten Ihre Anwendungsprogramme nicht mit den *Funktionsaufrufen* für die Standardeingabe oder Standardausgabe, so wird die Datenumleitung vom Programm nicht erkannt. Diese Funktion kann dann nicht in Verbindung mit dem Programm benutzt werden.

Ausgabe in eine bestehende Datei

Wird die Funktion der Datenumleitung benutzt, um Daten in eine Datei auszugeben, so wird diese Datei neu angelegt (falls noch nicht vorhanden) oder der Inhalt einer vorhandenen Datei überschrieben. Mit der hier beschriebenen Funktion ist es möglich, bei Datenumleitung, Daten an eine bestehende Datei anzuhängen. Dies geschieht für Dateien mit dem Zeichen:

Befehl >> Dateibez

Für entsprechende Peripheriegeräte lautet die Schreibweise:

Befehl >> Device

Da es nicht möglich ist, eine Ausgabe an ein Peripheriegerät anzuhängen, erzielt diese Funktion dasselbe Ergebnis wie die Funktion „>“.

Im vorherigen Abschnitt wurde als Beispiel die Datei DATEI.DIR mit der Datenumleitungsfunktion angelegt. Diese Datei enthält die Informationen des Inhaltsverzeichnisses. An diese Datei läßt sich mit der neuen Funktion „ >> “ ein weiteres Inhaltsverzeichnis anhängen. Geben Sie ein:

```
DIR D: >> DATEI.DIR
```

Befehlsverkettung und Filter

Filter

Ein *Filter* ist ein externer Befehl, der Daten vom Standardeingabegerät liest, diese Daten verändert und die veränderten Daten dann an das Standardausgabegerät wieder ausgibt. Die Daten werden somit *gefiltert*.

Die von MS-DOS zu Verfügung gestellten Filter können alleine benutzt werden oder auch in Verbindung mit anderen Befehlen. Sie ersetzen eine Vielzahl von Befehlen. Von MS-DOS werden folgende Filter zu Verfügung gestellt:

- **CIPHER** verschlüsselt oder entschlüsselt die Daten des Standardeingabegerätes
- **FIND** sucht nach einer vorgegebenen Zeichenkette im Standardeingabegerät
- **MORE** gibt die Daten des Standardeingabegerätes seitenweise auf dem Bildschirm aus
- **SORT** sortiert die Daten des Standardeingabegerätes und gibt diese dann zum Standardausgabegerät

Befehlsverkettung und Filter

Filter lassen sich sinnvollerweise mit anderen Befehlen verbinden. Sie lesen Daten, verändern diese und geben sie dann wieder aus. Ein Beispiel:

SORT < TRASH > TREASURE

Der Inhalt der Datei TRASH wird vom Filter SORT gelesen und anschließend sortiert in die Datei TREASURE wieder ausgegeben.

Arbeiten Ihre Anwendungsprogramme nicht mit den *Funktionsaufrufen* für die Standardeingabe oder Standardausgabe, so können Filter nicht von diesen Programmen benutzt werden.

Befehlsverkettung

Bei einer *Befehlsverkettung* wird die Standardausgabe des ersten Befehls zur Standardeingabe des zweiten Befehls. Die beiden Befehle werden durch das Zeichen (¦) miteinander verbunden. Dies geschieht in der Form:

Befehl1 ¦ Befehl2

Dabei wird immer die Ausgabe des Befehls auf der linken Seite des Balkens (¦) zur Eingabe des Befehls auf der rechten Seite des Balkens. So können mehrere Befehle miteinander *verkettet* werden.

Das geht bis zu einer maximalen Zeichenzahl von 127. Es können alle MS-DOS-Befehle miteinander verbunden werden, so lange sie die Standardeingabe und Standardausgabe benutzen.

Während der *Befehlsverkettung* werden von MS-DOS vorübergehend Dateien auf der Platte angelegt, die die Ein- und Ausgaben der einzelnen Befehle speichern. Diese Dateien werden im momentanen Inhaltsverzeichnis angelegt und haben als Bezeichnung die aktuelle Uhrzeit in hexadezimaler Schreibweise. Dadurch ist auch bei Multi-User-Betrieb gewährleistet, daß für jeden Bediener eigene Zwischendateien angelegt werden. Ein Beispiel:

0A300C18
0A300C1E

Für jede Befehlsverkettung werden mindestens zwei Dateien benötigt.

Normalerweise sehen Sie diese Dateien nicht, da Sie zum Schluß der Verkettung wieder gelöscht werden. Kann die Befehlszeile nicht korrekt abgearbeitet werden, ist es möglich, daß diese Dateien noch nicht wieder gelöscht sind.

Nachfolgend einige Beispiele für die Befehlsverkettung:

Wollen Sie das Inhaltsverzeichnis sortiert auf den Drucker ausgeben, geben Sie ein:

```
DIR C: ¦ SORT > PRN
```

Ist dieses Verzeichnis sehr umfangreich, kann es auf dem Bildschirm auch seitenweise ausgegeben werden. Die Befehlsfolge wird dann leicht verändert:

```
DIR C: ¦ SORT ¦ MORE
```

Ebenso ist es möglich, dieses sortierte Inhaltsverzeichnis in eine Datei zu geben, die nicht im momentanen Verzeichnis ist:

```
DIR C: ¦ SORT > \TEST\VERZ.TXT
```

Jetzt wird das sortierte Verzeichnis in die Datei mit dem Namen VERZ.TXT geschrieben. Diese Datei wird im Subdirectory TEST gespeichert.

Benutzt das Anwendungsprogramm die Standardeingabe um die Tastatur zu lesen, können Sie alle Befehle für das Programm in einer Textdatei speichern. Das Programm arbeitet dann die Textdatei ab und nimmt keine Eingaben von der Tastatur entgegen. Ein Beispiel:

```
EDLIN BEISPIEL.DOC < BEISPIEL.EIN
```

Bildschirmausgabe auf Drucker

Das Betriebssystem MS-DOS stellt verschiedene Möglichkeiten zur Verfügung, mit denen Sie den *Bildschirminhalt* auf einen Drucker ausgeben können. Sie können ein Protokoll sämtlicher Befehle auf dem Drucker erstellen. Sie können jederzeit den gesamten Bildschirminhalt, einschließlich Grafikzeichen, ausdrucken lassen. Für den Druck der Grafikzeichen werden besondere druckerspezifische Programme zur Verfügung gestellt.

Druckerprotokoll

Mit der Tastenkombination **CTRL-PRTSC** können Sie jederzeit das Betriebssystem anweisen, sämtliche Zeichen, die auf dem Bildschirm erscheinen, ebenfalls auf den Drucker auszugeben. Diese Tastenkombination arbeitet als *Schalter*. Bei einem erneuten Betätigen wird der Drucker wieder ausgeschaltet. Damit werden sämtliche Befehlsein- und Befehlsausgaben auf dem Drucker mitgeschrieben. Die Ausgabe erfolgt immer auf den ersten an das System angeschlossenen Drucker, der mit LPT1 oder PRN bezeichnet wird. Hierbei werden nur alphanumerische Zeichen gedruckt. Diese Funktion bleibt so lange eingeschaltet, bis sie mit der gleichen Tastenkombination wieder ausgeschaltet wird.

Ebenso ist es möglich diese Funktion zu jedem beliebigen Zeitpunkt mit der Tastenkombination **CTRL-P** ein- und auch wieder auszuschalten.

Bildschirminhalt drucken

Zusätzlich zur oben beschriebenen Funktion eines Protokolls auf dem Drucker ist es möglich, gezielt bestimmte Informationen auszudrucken. Es besteht jederzeit die Möglichkeit, den kompletten Bildschirminhalt auf den Drucker auszugeben. Durch Betätigen der Tastenkombination **SHIFT-PRTSC** wird das gerade ablaufende Programm angehalten und der Bildschirminhalt auf den Drucker gegeben. Es wird eine *Hardcopy* des Bildschirms erstellt. Ist diese Kopie fertig, ist der Druckvorgang beendet. Er wird erst wieder durch erneutes Betätigen der Tasten aktiviert. Diese Funktion ist auch in einem Anwenderprogramm aufrufbar, im Gegensatz zum Protokoll, daß nur auf Betriebssystemebene möglich ist.

Bildschirmausgabe auf den Drucker

Diese jederzeit aufrufbare Funktion kann allerdings nur alphanumerische Zeichen drucken. Haben Sie Ihren Bildschirm auf Grafik umgeschaltet, z. B. mit einem BASIC-Befehl, und es sind grafische Zeichen auf dem Bildschirm dargestellt, so werden diese nicht mit ausgegeben.

Wollen Sie auch grafische Zeichen ausdrucken, benötigen Sie zur Unterstützung des Druckers ein *Zusatzprogramm*. Dieses Zusatzprogramm wird für einige Drucker ebenfalls auf der Systemdiskette 1 mitgeliefert. Es wurde speziell für die Funktion der Bildschirmkopie geschrieben. Da es *druckerspezifisch* ist, gibt es für jeden Drucker ein eigenes Zusatzprogramm. Auf Ihrer Systemdiskette werden folgende Programme mitgeliefert:

- PSCHPTJ für Hewlett Packard Thinkjet Printer
- PSCIDS für IDS Prism Drucker
- PSCOKI für Okidata Drucker
- PSCTS315 für Transtar 315 Color Drucker
- PSCP920 für Printek 920 Drucker
- PSCMX80 für Epson MX-80 Drucker mit Grafikoption
- PSCMPI für MPI Drucker

Voraussetzung für diese Programme ist, daß sich der Bildschirm im Grafikmodus befindet. Die Programme werden folgendermaßen aufgerufen:

- Sie geben den Namen des Zusatzprogrammes ein, entsprechend dem angeschlossenen Drucker. Das Programm wird dann in den Arbeitsspeicher geladen.
- Rufen Sie Ihr Arbeitsprogramm auf. Achten Sie darauf, daß der Bildschirm im Grafikmodus ist. Immer wenn Sie jetzt eine Kopie des Bildschirms benötigen, betätigen Sie die Tasten **SHIFT-PRTSC** gleichzeitig. Der komplette Inhalt des Schirmes wird jetzt inklusive aller Grafik- und Sonderzeichen abgedruckt.
- Ist der Druckvorgang beendet, kann er jederzeit wieder durch die Tastenkombination aufgerufen werden.

Bildschirmausgabe auf den Drucker

Dieses Unterstützungsprogramm ist im Speicher weiterhin vorhanden, selbst wenn Sie den Grafikmodus verlassen. Sie können dann immer noch sämtliche alphanumerischen Zeichen ausdrucken. Kehren Sie in den Grafikmodus zurück, werden wieder sämtliche sichtbaren Zeichen gedruckt.

Die PSC-Zusatzprogramme arbeiten nur, wenn sich der Bildschirm im Grafikmodus befindet. Sie können vom Textmodus in den Grafikmodus umschalten, indem Sie entsprechende Funktionen des Monitorprogrammes benutzen. Dies wird ausführlich im Bedienerhandbuch Ihres PC beschrieben.

Unterbrechen Sie das laufende Programm durch die Tastenkombination **CTRL-ALT-RETURN**. Es erscheint das Prompt des Monitors (-->). Geben Sie jetzt folgenden Befehl ein:

```
VM6
```

Der Bildschirm wird umgeschaltet und gelöscht. Mit der Eingabe von

```
G
```

kehren Sie in das laufende Programm zurück. Alle Daten des Programmes bleiben dabei erhalten. Mit der Eingabe von

```
VM3
```

schaltet das Monitorprogramm in den Textmodus zurück.

Zusammenfassung

Im Betriebssystem MS-DOS werden sämtliche Ein-/Ausgabefunktionen durch Transportieren von Daten zwischen Peripheriegeräten (Devices) vollführt. Diese Arbeit wird von Charakter- und Blocktreibern ausgeführt, die die Informationen vom MS-DOS I/O Handler zum Gerät schicken. Die interne Verbindung zwischen den Geräten und dem I/O Handler wird unterstützt durch die Standardeingabe und die Standardausgabe.

Das Standardeingabegerät des Computers ist die Tastatur. Das Standardausgabegerät des Computers ist der Bildschirm. Es können mit speziellen Funktionen diese Geräte zeitweilig neu definiert werden. Dies kann durch Datenumleitung oder Verkettung von Befehlen geschehen. Damit ist eine Eingabe auch aus einer Datei oder von einem anderen Gerät möglich. Ebenso ist die Ausgabe in eine Datei oder zu einem anderen Gerät möglich. Als neues Eingabegerät darf nur ein schreibendes Gerät definiert werden, als Ausgabegerät nur ein lesendes Gerät.

MS-DOS unterstützt ebenfalls verschiedene Möglichkeiten der Druckerausgabe. Es kann die laufende Ein-/Ausgabe der Befehle auf Betriebssystemebene ausgedruckt werden. Ebenso kann eine Hardcopy des Bildschirmes erstellt werden. Durch eine einfache Tastenkombination kann der gesamte Bildschirminhalt auf den Drucker gegeben werden. Sollen Grafik- oder Sonderzeichen mit ausgedruckt werden, muß zur Druckerunterstützung ein Hilfsprogramm aufgerufen werden. Dieses Programm ist druckerspezifisch. Es ermöglicht, einmal geladen, jederzeit den Ausdruck von grafischen Zeichen.

Kapitel 9

Systembestandteile

Übersicht

MS-DOS ist ein *Betriebssystem,* das aus verschiedenen Teilen besteht. Jedes Teil hat seine besondere Funktion sowie seinen bestimmten Speicherplatz auf der Diskette und im Arbeitsspeicher. Durch diese Plätze werden die einzelnen Teile des Systems identifiziert.

In diesem Kapitel wird jedes *Teil* (Datei) des Betriebssystems ausführlich erklärt. Es wird erklärt, welche Funktionen die einzelnen Teile haben und wie sie ausgeführt werden. Dieses Wissen hilft Ihnen, die unsichtbaren Vorgänge in Ihrem Computer besser zu verstehen.

Dieses Kapitel erklärt Ihnen zu Anfang aus welchen Teilen generell ein Betriebssystem besteht und welche Teile von MS-DOS diese Funktionen ausführen.

Allgemeine Systembestandteile

Abgesehen von seinen umfangreichen Dienstprogrammen hat MS-DOS mit vielen anderen *Betriebssystemen* für Mikrocomputer folgende fünf Gemeinsamkeiten:

- Boot-Loader — lädt die restlichen Systemdateien in den Arbeitsspeicher, wird danach nicht mehr benötigt
- Ein-/Ausgabe Verwaltung — steuert die Ein- und Ausgabe
- Datei-verwaltung — steuert das Dateisystem
- Speicher-verwaltung — steuert den Arbeitsspeicher
- ausführendes Teil — lädt Programme und veranlaßt den Prozessor, die dort gespeicherten Anweisungen abzuarbeiten.

Um diese fünf Funktionen auszuführen, besteht das MS-DOS Betriebssystem aus den vier Teilen:

- BOOT-LOADER
- IO.SYS
- MSDOS.SYS
- COMMAND.COM

Boot-Loader

Der Boot-Loader ist ein eigenes kleines *Programm,* das den Rest des Betriebssystemes von der Platte in den Arbeitsspeicher kopiert (bootet). Der *Boot-Loader* befindet sich immer auf dem ersten Sektor eines externen Speichermediums. Er muß alle Daten enthalten, um den Rest des Systems zu laden.

Wurde der Rest des Betriebssystemes in den Arbeitsspeicher kopiert, ist der Boot-Loader bei den meisten Systemen nicht mehr erforderlich. Er wird dann von anderen Programmen überschrieben.

Die Funktion des Boot-Loader wird bei MS-DOS von einem Teil übernommen, der ebenfalls BOOT-LOADER genannt wird.

Ein-/Ausgabeverwaltung

Das Ein-/Ausgabeteil steuert den *Informationsfluß* innerhalb des Systemes. Es weiß, welche Geräte an den Computer angeschlossen sind und wo. Mit diesen Informationen und Anweisungen des ausführenden Teiles versehen, steuert es die Informationen zwischen den einzelnen Geräten. Damit ist jedes angeschlossene Gerät in der Lage, die richtigen Daten zu empfangen und zu senden.

Die Funktionen des Ein-/Ausgabeteiles werden bei MS-DOS von den Teilen IO.SYS und MSDOS.SYS übernommen.

Dateiverwaltung

Das Dateiverwaltungsteil bearbeitet alle Befehle zur *Dateimanipulation.* Es stellt Daten von Bytes zu einer Gruppe zusammen, damit diese auf die Platte geschrieben werden können. Dabei ordnet es auf der Platte die Gruppen zu einer *Datei.* Es verarbeitet die Informationen über die Dateien. Das Teil weiß, wo eine Datei auf der Platte gespeichert ist und benennt diese Datei. Jedesmal wenn sich eine neue Diskette im Laufwerk befindet, wird diese vom Dateiverwaltungsteil gelesen, damit die Namen und weiteren Daten der neuen Dateien bekannt sind.

Allgemeine Systembestandteile

Die vom Dateiteil verwalteten Daten bleiben auf der Diskette gespeichert. Da die Diskette ein externer Speicher ist, sind die Daten immer vorhanden. Sie gehen nicht verloren, wenn das System ausgeschaltet wird.

Die Funktionen zur Dateiverwaltung werden bei MS-DOS vom Teil MSDOS.SYS übernommen.

Speicherverwaltung

Die *Speicherverwaltung* teilt den Arbeitsspeicher (RAM) auf. Dem Betriebssystem wird ein genau festgelegter Teil zugewiesen, der von keinem anderen Programm überschrieben werden darf. Der Platz für Anwenderprogramme muß reserviert werden, genauso wie der Speicherplatz für den Datentransfer. Werden weitere Programme aufgerufen, wird von diesem Teil entschieden, welche Programme im Arbeitsspeicher gelöscht werden und welche nicht.

Die vom Speicherteil verwalteten Daten sind nach einem Stromausfall oder einem neuen Systemstart nicht mehr vorhanden. Es sind flüchtige Daten.

Die Funktionen zur Speicherverwaltung werden bei MS-DOS vom Teil MSDOS.SYS übernommen.

Programmausführung

Das Teil zur *Programmausführung* überwacht das gesamte Betriebssystem. Es stellt die Verbindung zwischen dem Benutzer und dem Computer her. Es interpretiert die Befehle und leitet sie an die entsprechenden Teile des Betriebssystemes weiter. Es überwacht den Programmaufruf, indem es kontrolliert, ob das richtige Programm geladen wird. Ebenso antwortet es auf die Eingaben des Benutzers. Geben Sie einen Befehl zur Programmausführung ein, so veranlaßt dieser Teil folgende Aktivitäten:

- das Programm wird gesucht
- das Programm wird in den Arbeitsspeicher geladen
- das Programm beginnt mit der Befehlsausführung

Die Funktionen zum Programmaufruf werden bei MS-DOS vom Teil COMMAND.COM übernommen.

MS-DOS Bestandteile

Das Betriebssystem MS-DOS besteht aus den folgenden vier Teilen:

- BOOT-LOADER; ist Geräteabhängig
- IO.SYS; ist Geräteabhängig
- MS-DOS.SYS; ist Geräteunabhängig
- COMMAND.COM; ist Geräteunabhängig

Der folgende Abschnitt beschreibt die Eigenschaften der einzelnen Teile. Da die letzten drei Teile als Dateien auf der Platte vorhanden sind, werden sie mit ihren Dateinamen beschrieben.

BOOT-LOADER

Der BOOT-LOADER ist ein Programm, daß zu Beginn des Systemstarts als erstes Programm geladen wird. Es lädt dann die Datei IO.SYS von einem externen Speicher in den Arbeitsspeicher des Computers. Das Programm bootet das System. Dieses Programm ist nicht als eigene Datei vorhanden, sondern auf der Diskette oder Winchester im ersten Sektor, Spur 0 der Seite 0 enthalten. Der BOOT-LOADER wird durch das Programm FORMAT bei jedem Formatieren einer neuen Diskette dorthin kopiert.

Der BOOT-LOADER enthält alle wichtigen Informationen über die Diskette, die Hardware des Systemes und den Speicherplatz der Systemdateien. Wenn dieses Programm die Datei IO.SYS geladen hat, übergibt es ihr die weitere Kontrolle. Es wird dann nicht mehr benötigt. Dieses Programm ist hardwareabhängig.

MS-DOS Bestandteile

IO.SYS

Die Datei IO.SYS ist die *Arbeitsdatei* des Systemes. Sie verarbeitet alle Befehle hardwaremäßig. Sie kennt die Eigenschaften der angeschlossenen Geräte und weiß wie die Befehle dort verarbeitet werden. Die Datei ist hardwareabhängig.

IO.SYS erhält die allgemeinen Befehle der Datei MSDOS.SYS. Die Datei übersetzt diese Befehle entsprechend der angeschlossenen Hardware, damit sie dort richtig verarbeitet werden können.

Zum Beispiel gibt die Datei MSDOS.SYS der Datei IO.SYS den Befehl, Daten an das Peripheriegerät mit dem Namen CON auszugeben. Die Datei IO.SYS weiß jetzt, daß mit dem Namen CON das Standardausgabegerät (Bildschirm) und das Standardeingabegerät (Tastatur) bezeichnet ist. Die Datei IO.SYS gibt jetzt die Daten auf dem Bildschirm aus. Dabei berücksichtigt die Datei die spezifischen Daten des Bildschirmgerätes, wie Übertragungsgeschwindigkeit, Zeichendarstellung und Anzahl der Zeilen und Spalten auf dem Schirm.

Da diese Daten bei jedem Gerät anders sind, muß die Datei für jeden Computer neu geschrieben werden.

MSDOS.SYS

Die Datei MSDOS.SYS ist der *Leiter* des ganzen Systemes. Sie gibt die Befehle, wenn eine Arbeit auszuführen ist. Diese Befehle bezeichnen nicht die Geräte, die die Arbeit übernehmen sollen. Sie enthalten auch keine Anweisungen über die Art der Ausführung. Daher ist die Datei hardwareunabhängig und kann auf jedem Gerät eingesetzt werden.

MSDOS.SYS stellt eine intelligente Verbindung für Anwenderprogramme dar. Es verfügt über Programmteile zur Dateiverwaltung, zur Steuerung von Informationsblöcken bei Plattenzugriff und einer Reihe von nützlichen Routinen zur Unterstützung der Anwenderprogramme. Die Datei kann Anrufe der Anwenderprogramme erkennen und zur Datei IO.SYS weiterleiten, die diese Anweisungen dann ausführt.

Die Datei MSDOS.SYS erledigt sämtliche Arbeiten zur Dateiverwaltung. Ebenso führt sie einige Ein- und Ausgaben durch.

COMMAND.COM

Die Datei COMMAND.COM stellt die *Verbindung* zwischen dem Benutzer und dem Computer her. Die Datei nimmt die Befehle des Benutzers entgegen und interpretiert diese. Sie wird daher auch als *Befehlsinterpreter* bezeichnet. Ab der Version 2.0 ist es möglich, diese Datei durch einen anderen Interpreter zu ersetzen. Dazu muß die Datei CONFIG.SYS einen entsprechenden Befehl enthalten.

Die Datei COMMAND.COM besteht aus drei Teilen:

- dem Initialisierungsteil
- dem beständigen Teil
- dem überschreibbaren Teil

Der folgende Abschnitt beschreibt die einzelnen Teile.

Der Initialisierungsteil

Dieser Teil wird oberhalb des nicht überschreibbaren Teiles in den Arbeitsspeicher geladen. Er übernimmt die *Kontrolle* während des Systemstarts. Dieser Teil sucht die niedrigste freie Speicheradresse, ab der ein Anwenderprogramm geladen werden kann. Hat es diese Funktion erfüllt, wird es nicht mehr benötigt. Das erste zu ladende Programm überschreibt diesen Teil.

Der beständige Teil

Dieser Teil der Datei COMMAND.COM wird beim Systemstart in den Arbeitsspeicher geladen und bleibt dort so lange, bis das System neu gestartet wird. Er enthält die nötigen Programmteile, die erforderlich sind, um Programme zu laden. Er bearbeitet Fehler, die von den Peripheriegeräten verursacht werden. Hier wird ebenfalls die Programmunterbrechung durch die Tastenkombination CTRL-BREAK bearbeitet. Ebenso enthält dieser Teil ein Programm zum Nachladen des überschreibbaren Teiles.

Der überschreibbare Teil

Dieser Teil der Datei COMMAND.COM enthält alle internen Befehle. Er gibt das Systemanforderungszeichen aus. Hier befindet sich ebenfalls der Befehlsinterpreter und ein Verarbeitungsprogramm für Batchdateien. Dieser Teil wird an das obere Ende des verfügbaren Speichers geladen. Sollte ein externer Befehl oder ein Anwenderprogramm viel Arbeitsspeicher benötigen, kann dieser Teil der Datei COMMAND.COM überschrieben werden. Nach Rückkehr zum Betriebssystem wird dieser Teil dann automatisch wieder nachgeladen. Der überschreibbare Teil der Datei COMMAND.COM kann überschrieben werden, weil die dort vorhandenen Funktionen während der Abarbeitung eines anderen Programmes nicht benötigt werden. Diese Programme sind so geschrieben, daß sie alleine ablaufen können. Sie überschreiben diesen Teil erst dann, wenn der restliche Arbeitsspeicher schon belegt ist und zur Ausführung des Programmes weitere Speicherplätze belegt werden müssen.

Speicherung auf der Diskette

Die einzelnen Teile des Betriebssystemes MS-DOS können mit den externen Befehlen FORMAT oder SYS auf die Diskette oder Winchester kopiert werden. Dies geschieht in der Reihenfolge:

BOOT-LOADER
IO.SYS
MSDOS.SYS
COMMAND.COM

Das Programm BOOT-LOADER wird mit dem Befehl FORMAT immer automatisch übertragen. Die anderen Teile werden nur bei Eingabe des Zusatzes **/S** kopiert.

FORMAT schreibt das Programm BOOT-LOADER immer auf den ersten Sektor (absolute Adresse: Seite 0, Spur 0, Sektor 1) einer Diskette oder eines Winchesterlaufwerkes.

Speicherung auf der Diskette

Die folgenden Sektoren werden zur *internen Verwaltung* der Diskette benötigt und können von Ihnen nicht beschrieben werden. Diese Sektoren enthalten eine Tabelle mit den Speicherplätzen der Dateien, die auf dieser Diskette gespeichert sind (werden). Diese Tabelle nennt man *File Allocation Table* (FAT). Die Tabelle ist aus Sicherheitsgründen zweimal vorhanden. Auf den nächsten Sektoren ist das Inhaltsverzeichnis (Rootdir) gespeichert.

Nur wenn die ersten beiden Dateien auf einer Diskette die Dateien IO.SYS und MSDOS.SYS sind, ist es möglich, von dieser Diskette zu booten. Sind die Dateien auf der Diskette, werden sie als Systemdateien markiert, die im Inhaltsverzeichnis nicht offen angezeigt werden. Dadurch ist es nicht möglich, diese Dateien zu überschreiben oder anders zu benennen.

Findet der BOOT-LOADER die Datei IO.SYS nicht am angegebenen Platz, gibt er eine Fehlermeldung aus und bricht den Bootvorgang ab. Es erscheint die Meldung:

No system	Kein System

Die Datei COMMAND.COM dagegen, kann an jede beliebige Stelle auf der Diskette kopiert werden. Sie ist auch nicht als Systemdatei oder als geschützte Datei markiert. Daher wird sie im Verzeichnis immer angezeigt.

Ist die Datei COMMAND.COM oder ein anderer Befehlsinterpreter, der in der Datei CONFIG.SYS benannt wird, nicht auf der Diskette vorhanden, erscheint die Fehlermeldung:

Bad or missing Command Interpreter	Kein Befehlsinterpreter vorhanden

Der Befehl SYS kopiert nur drei Bestandteile des Betriebssystemes auf eine formatierte Diskette. Es sind dies die verdeckten Dateien:

BOOT-LOADER
IO.SYS
MSDOS.SYS

Der Befehl kennzeichnet die beiden Dateien IO.SYS und MSDOS.SYS immer als verdeckte Dateien.

Die Datei COMMAND.COM wird mit diesem Befehl nicht kopiert, da Sie nicht als Systemdatei behandelt wird.

Speicherung im Arbeitsspeicher

Die einzelnen Teile des Betriebssystemes MS-DOS werden an genau festgelegten Stellen in den *Arbeitsspeicher* (RAM) kopiert. Dort bleiben sie solange, bis Sie das System neu booten. Das folgende Bild ist ein Speicherauszug, der den Zustand nach dem Bootvorgang zeigt. Dies ist der Zeitpunkt, an dem das Prompt erscheint.

Es werden die Speicherplätze mit den niedrigen Adressen als untere Plätze bezeichnet und analog dazu die Plätze mit den hohen Adressen als obere Plätze. Dabei ist die obere Grenze vom Speicherausbau abhängig.

Tabelle der Unterbrechungsvektoren
IO.SYS - Verbindung zwischen MSDOS.SYS und Hardware
MSDOS.SYS - MS-DOS Vektorenverarbeitung, Unterprogramme
MS-DOS Gerätetreiber, Zwischenspeicher
Nicht überschreibbarer Teil von COMMAND.COM
Freier Speicher für Anwenderprogramme
Überschreibbarer Teil von COMMAND.COM

Bild 9.1
Speicherplätze der Systembestandteile

Der Bootvorgang

Während des *Bootvorganges* laufen verschiedene Schritte nacheinander unsichtbar ab. Die einzelnen Teile Ihres Computersystemes beeinflussen sich wechselseitig und ermöglichen Ihnen zum Schluß des Vorganges, Befehle einzugeben. Dann können Sie mit dem Computer arbeiten.

Schritt 1: Der Bootvorgang beginnt (s. a. Kapitel 6).

Schritt 2: Booten Sie von der Diskette, wird der weitere Ablauf ab Schritt 5 erklärt.

Schritt 3: Das Monitorprogramm lädt den ersten Sektor der Winchester in den Speicher und übergibt ihm die Kontrolle (s. a. Kapitel 18).

Schritt 4: Das Bootprogramm der Winchester lädt jetzt vom hier bezeichneten Startlaufwerk das Programm BOOT-LOADER. Dieses übernimmt die weitere Kontrolle.

a. Enthält das Startlaufwerk MS-DOS, wird, wie mit Schritt 6 beschrieben, fortgefahren.

b. Enthält das Startlaufwerk ein anderes Betriebssystem, so übernimmt dieses jetzt die Kontrolle.

Schritt 5: Das Monitorprogramm lädt vom ersten Sektor der Diskette den BOOT-LOADER und übergibt ihm die Kontrolle.

Schritt 6: Der BOOT-LOADER lädt die Datei IO.SYS und übergibt ihr die Kontrolle.

Der Bootvorgang

Schritt 7: IO.SYS initialisiert die angeschlossenen Geräte und lädt die Datei MSDOS.SYS.

Schritt 8: IO.SYS übergibt die Kontrolle an das Modul SYSINIT. Dieses Modul stellt die Verbindung zwischen IO.SYS und MSDOS.SYS her.

Schritt 9: SYSINIT kopiert sich selber an einen anderen Platz im Arbeitsspeicher.

Schritt 10: SYSINIT kopiert die Datei MSDOS.SYS an einen anderen Platz im Arbeitsspeicher.

Schritt 11: SYSINIT startet das Programm MSDOS.SYS.

Schritt 12: SYSINIT sucht die Datei CONFIG.SYS

a. Ist die Datei CONFIG.SYS vorhanden, wird der Teil MSDOS.SYS entsprechend den dort vorhandenen Befehlen modifiziert.

b. Ist die Datei CONFIG.SYS nicht vorhanden, wird, wie mit Schritt 14 beschrieben, fortgefahren.

Schritt 13: SYSINIT durchsucht das Verzeichnis nach dem Befehlsinterpreter. Diese Datei ist mit dem Befehl SHELL näher bezeichnet. Fehlt der Befehl in der Datei CONFIG.SYS, ist der Befehlsinterpreter automatisch die Datei COMMAND.COM.

a. Ist der Interpreter die Datei COMMAND.COM und kann diese geladen werden, wird ihr anschließend die weitere Kontrolle übergeben.

b. Wird der bezeichnete Befehlsinterpreter nicht gefunden, erscheint die Fehlermeldung:

Bad or missing Command Interpreter	Der Befehlsinterpreter fehlt oder ist falsch

c. Wird ein anderer Interpreter in der Datei CONFIG.SYS bezeichnet und kann dieser geladen werden, wird ihm die weitere Kontrolle übergeben.

Schritt 14: COMMAND.COM sucht die Datei AUTOEXEC.BAT.

a. Ist die Datei vorhanden, werden die dort gespeicherten Befehle abgearbeitet.

b. Ist die Datei nicht vorhanden, werden Datum und Zeit abgefragt.

Anschließend erscheint der Programmname auf dem Bildschirm:

```
MS-DOS Version 3.10
Copyright 1981,82,83,85 Microsoft Corp.
```

Schritt 15: COMMAND.COM schreibt das Prompt auf den Bildschirm und wartet auf Ihre Eingaben.

Befehlsverarbeitung

Sind alle Teile des Betriebssystemes MS-DOS in den Arbeitsspeicher geladen, können Sie Ihre *Befehle* eingeben. Im folgenden Abschnitt ist erklärt, welche Vorgänge zwischen der Befehlseingabe und dem Beginn der Befehlsausführung ablaufen.

Schritt 1: Sie geben den Befehl nach dem Prompt ein.

a. Wird der Befehl von Ihnen eingegeben, ist die Eingabe mit **RETURN** beendet.

b. Wird der Befehl von einer Batchdatei gelesen, ist die Eingabe beendet, wenn das Zeichen **RETURN** (unsichtbar) gelesen wird.

Schritt 2: Die komplette Befehlszeile wird zum Interpreter gegeben.

Schritt 3: Der Interpreter untersucht die Zeile Zeichen für Zeichen und vergleicht sie mit den gespeicherten internen Befehlen.

a. Ist in der Zeile ein interner Befehl vorhanden, wird dieser ausgeführt.

b. Ist kein interner Befehl vorhanden, wird wie mit Schritt 4 beschrieben fortgefahren.

Schritt 4: Es erfolgt auf der Platte das Absuchen des bezeichneten Inhaltsverzeichnisses nach dem externen Befehl oder Programmnamen.

a. Ist im Verzeichnis eine einzelne Datei mit dem Namen vorhanden (der Zusatz muß .COM, .EXE oder .BAT sein), lädt der Computer sie, und es beginnt die Abarbeitung.

b. Sind im Verzeichnis mehrere Dateien mit dem vorgegebenen Namen vorhanden, wird nur eine davon geladen und ausgeführt. Die zu ladende Datei wird dabei nach folgenden Regeln bestimmt:

1. Eine Datei mit dem Zusatz .COM wird immer zuerst geladen und abgearbeitet. Die Dateien mit dem Zusatz .EXE oder .BAT werden dabei nie berücksichtigt.

2. Eine Datei mit dem Zusatz .EXE wird immer vor einer Datei mit dem Zusatz .BAT geladen und abgearbeitet. Die ebenfalls vorhandene Batchdatei würde so nie aufgerufen.

c. Wird der eingegebene Name nicht gefunden, erscheint folgende Fehlermeldung:

Bad command or file name	Befehl oder Datei falsch

und es erscheint wieder das Prompt.

Geben Sie in der Befehlszeile einen *Funktionsnamen* ein, können Sie damit vier verschiedene Funktionen bezeichnen. Das kann ein interner oder ein externer Befehl sein. Der externe Befehlsname kann den Namenszusatz .COM oder .EXE haben. Sie können damit auch eine Batchdatei (Zusatz .BAT) bezeichnen. Trifft jetzt der Funktionsname auf mehrere oder alle verschiedenen Funktionen zu, muß das Betriebssystem entscheiden, welche Funktion zuerst auszuführen ist.

Die Datei COMMAND.COM entscheidet, welche Funktion auszuführen ist. Es wird immer die Funktion mit der höchsten Priorität ausgeführt:

1. interner Befehl
2. externer Befehl mit dem Namenszusatz .COM
3. externer Befehl mit dem Namenszusatz .EXE
4. externer Befehl mit dem Namenszusatz .BAT

Befehlsverarbeitung

Geben Sie zum Beispiel folgende Befehlszeile ein:

```
PROGRAM C:THISFILE.OBJ /K /R
```

Dann wird vom Interpreter die Zeile Zeichen für Zeichen abgesucht und festgestellt, daß der Funktionsname „PROGRAM" ist. Jetzt wird dieses Wort mit allen internen Befehlen verglichen:

BREAK	DIR	MKDIR	RMDIR
CD	ECHO	PATH	SET
CHDIR	ERASE	PAUSE	SHIFT
CLS	EXIT	PROMPT	TIME
COPY	FOR	RD	TYPE
CTTY	GOTO	REM	VER
DATE	IF	REN	VERIFY
DEL	MD	RENAME	VOL

Da der Name „PROGRAM" mit keinem der vorstehenden Befehle übereinstimmt, sucht der Interpreter auf der Platte nach einer Datei mit dem Namen.

Werden vom Interpreter auf der Platte mehr als eine Datei gefunden, deren erster Name „PROGRAM" ist, so wird nur die Datei mit der höchsten *Priorität* geladen. Die Tabelle 9.1 zeigt verschiedene Möglichkeiten. Alle anderen Dateien würden nie erfaßt. Sie sollten daher diese Dateien entweder umbenennen oder auf verschiedene Laufwerke kopieren.

Findet der Interpreter die eingegebene Funktion nicht, weder bei den internen Befehlen noch als Datei auf der Platte, erscheint die Fehlermeldung:

```
Bad command or file name          Befehl oder Datei falsch
```

In der folgenden Zeile wird das Prompt ausgegeben.

Tabelle 9.1 Priorität der externen Befehle

Namensgleiche Dateien des Laufwerkes	von COMMAND.COM zur Bearbeitung ausgewählte Datei
PROGRAM.COM	PROGRAM.COM
PROGRAM.EXE	PROGRAM.EXE
PROGRAM.BAT	PROGRAM.BAT
PROGRAM.COM PROGRAM.EXE	PROGRAM.COM
PROGRAM.EXE PROGRAM.BAT	PROGRAM.EXE
PROGRAM.COM PROGRAM.EXE PROGRAM.BAT	PROGRAM.COM

Modifizieren der Datei MSDOS.SYS

Die Datei MSDOS.SYS kann nach Ihren speziellen *Wünschen* erweitert werden. Das geschieht mit Hilfe der Datei CONFIG.SYS.

Die Datei *CONFIG.SYS* ist eine Textdatei, in der Befehle gespeichert sind. Sie kann mit der Datei AUTOEXEC.BAT verglichen werden. Während des Systemstarts wird automatisch das Inhaltsverzeichnis des Startlaufwerkes nach dieser Datei abgesucht. Ist sie vorhanden, werden die dort gespeicherten Befehle ausgeführt.

Mit diesen Befehlen rufen Sie spezielle Funktionen auf, laden Sie weitere Gerätetreiber, reservieren Sie Zwischenspeicher für Ein-/Ausgabeoperationen mit der Platte, und Sie können auch einen anderen Befehlsinterpreter als die Datei COMMAND.COM bestimmen. Mit diesen Modifikationen sind Sie in der Lage, schnell und ohne Aufwand das Betriebssystem Ihren Wünschen anzupassen.

Die Datei CONFIG.SYS kann folgende Befehle verarbeiten:

BREAK = ON ¦ OFF

Dieser Befehl modifiziert die *Programmunterbrechnung* durch die Tastenkombination **CTRL-BREAK.** Ist die Funktion eingeschaltet, wird das laufende Programm sofort nach Betätigen der Tasten unterbrochen. Ist die Funktion ausgeschaltet, wird das Programm nur noch bei einer Ein- oder Ausgabe von einem Peripheriegerät unterbrochen. Ohne Modifizierung ist die Funktion ausgeschaltet.

BUFFERS = Nummer

Nummer ist eine dezimale Zahl zwischen 01 und 99 und legt die Anzahl der Zwischenspeicher fest, die im RAM zusätzlich (zwei sind generell vorhanden) für Daten reserviert werden. Dadurch wird vielfach der Programmablauf schneller. Jeder Zwischenspeicher (Buffer) belegt 528 Bytes des Arbeitsspeichers. Je nach benutztem Programm und Ausbau des Arbeitsspeichers ist hier eine Eingabe von 6 bis 20 Buffer sinnvoll.

COUNTRY = Nummer

Nummer ist eine dezimale Zahl zwischen 1 und 99 und bestimmt den landesspezifischen Datensatz, der nachgeladen werden kann. Dieser Datensatz enthält Angaben über das Format für die Datumsanzeige und die Uhrzeit. Ebenso ist das Währungssymbol gespeichert und weitere Angaben für das Formatieren von Ausgaben auf dem Bildschirm oder Drucker. Es ist somit möglich, das Format für Uhrzeit und Datum auf gewohntes Aussehen zu verändern. Der deutsche Datensatz ist mit der *Nummer* **49** aufzurufen, der amerikanische mit der Nummer 1. Wird der Befehl nicht eingegeben, ist automatisch der amerikanische Datensatz geladen. Die folgende Tabelle zeigt mögliche Eingaben.

Tabelle 9.2 Landeskennziffern

LAND	Kennziffer
Australien	061
Belgien	032
Dänemark	045
Deutschland	049
Finnland	358
Frankreich	033
Großbritanien, UK	044
Großbritanien, US	044
Israel	972
Italien	039
Kanada, EL	001
Kanada, SE	001
Niederlande	031
Norwegen	047
Schweiz, DT	041
Schweiz, FR	041
Spanien	034
USA	001

DEVICE = Dateiname

Dieser Befehl ruft eine *Datei* (Device Driver) auf, die als Erweiterung von MSDOS.SYS geladen wird. Diese Datei enthält das Steuerprogramm (Device Driver) für ein Peripheriegerät (Device). Damit wird der Datenfluß zwischen der Zentraleinheit und dem Gerät überwacht und falls erforderlich entsprechend verändert. Zwei Dateien werden bereits auf der Systemdiskette mitgeliefert. Dies sind die Dateien ANSI.SYS für die erweiterten Bildschirm- und Tastaturfunktionen und die Datei VDISK.SYS, die im Arbeitsspeicher ein Laufwerk simuliert (RAM-Disk). Beide Dateien werden am Ende dieses Kapitels näher beschrieben.

FCBS = Nummer1, Nummer2

Einige ältere Anwendungsprogramme benutzen zur Dateiverwaltung einen *FCB* (File Control Block). Sie bestimmen mit dem Wert für *Nummer1* die maximale Zahl der gleichzeitig durch einen FCB geöffneten Dateien. Der Wert für *Nummer2* bestimmt die Zahl der geschützten FCBs.

Versucht ein Anwendungsprogramm mehr FCBs zu öffnen als mit *Nummer1* bestimmt wurde, schließt DOS automatisch den zuletzt geöffneten FCB. Dies ist erforderlich, da der benutzte Speicherplatz benötigt wird. Mit dem Wert für *Nummer2* bestimmen Sie die Zahl der FCBs, die unbedingt offengehalten werden müssen.

Ohne Eingabe des Befehles wird Platz für vier FCBs reserviert. Davon ist kein FCB geschützt. Für Nummer1 können beliebige Werte im Bereich 1 bis 255 eingegeben werden. Für Nummer2 können beliebige Werte im Bereich 0 bis 255 eingegeben werden. Dabei darf die Zahl für Nummer2 nicht größer sein als für Nummer1. Beziehen sich zwei verschiedene FCBs auf dieselbe Datei, zählt MS-DOS dies als einen FCB. Sind die Werte für Nummer1 und Nummer2 gleich, kann MS-DOS keine weitere Datei öffnen und gibt eine entsprechende Fehlermeldung aus. Sie sollten dann den Wert für Nummer1 erhöhen.

Haben Sie Programme für Netzwerk installiert, und ein Programm versucht eine größere Zahl von FCBs zu öffnen als mit Nummer1 bestimmt, schließt MS-DOS den zuletzt geöffneten FCB wieder. Dies können Sie verhindern, indem Sie mit Nummer2 eine bestimmte Anzahl von FCBs schützen. MS-DOS erkennt diese FCBs dann nicht mehr als zuletzt geöffnet.

Versucht Ihr Programm eine Datei zu bearbeiten, dessen FCB zwangsweise geschlossen wurde, erscheint die Fehlermeldung:

FCB not available FCB nicht mehr vorhanden

Benutzt Ihr Programm keine FCBs sondern Dateinummern, benötigen Sie diesen Befehl nicht.

FILES = Nummer

Nummer ist eine dezimale Zahl zwischen 05 und 99 und legt die Anzahl der gleichzeitig zu benutzenden Dateien fest. Damit benötigt MS-DOS keinen FCB mehr zur Dateiverwaltung, sondern nur noch eine Dateinummer. Dieses System wurde mit der Version 2 eingeführt. Geben Sie den Befehl nicht ein, legt MS-DOS den Wert mit 08 fest. Eine Zahl kleiner als 05 wird nicht beachtet. Das Betriebssystem benötigt schon zur internen Verwaltung diese fünf Dateien, somit können drei Dateien von Anwenderprogrammen benutzt werden. Sämtliche Ein- und Ausgabeoperationen des Betriebssystemes werden mit Hilfe dieser fünf Dateien ausgeführt. Die folgende Tabelle gibt eine Übersicht.

Tabelle 9.3 Namen der von MS-DOS standardmäßig genutzen Dateien

NUMMER	NAME	FUNKTION	BEMERKUNGEN
0000	STDIN	Standardeingabe	kann umgeleitet werden
0001	STDOUT	Standardausgabe	kann umgeleitet werden
0002	STDERR	Fehlerausgabe	kann nicht umgeleitet werden
0003	STDAUX	Ausgabe auf	kann nicht umgeleitet
0004	STDPRN	Drucker od. Modem	werden

LASTDRIVE = Buchstabe

Sie bestimmen mit diesem Befehl die Zahl der benutzbaren logischen Laufwerke. Da unter MS-DOS Laufwerkbezeichnungen mit einem Buchstaben vorgenommen werden, bestimmt die Eingabe des Buchstabens das letzte gültige Laufwerk. Zum Beispiel können Sie mit dem neuen Programm SUBST weitere logische Laufwerke bestimmen. Ohne Eingabe des Befehles ist E: das letzte gültige Laufwerk. Bestimmen Sie eine kleinere Zahl als physikalisch vorhanden, wird der Befehl ignoriert.

SHELL = Dateiname

Dieser Befehl ruft eine Datei auf, die anstatt der Datei COMMAND.COM geladen wird. Diese Datei übernimmt dann die Funktion des *Befehlsinterpreters.* Dies kann eine beliebige Binärdatei (Zusatz COM) sein, aber niemals eine Datei mit dem Zusatz EXE.

ANSI.SYS

Die Datei ANSI.SYS ist ein funktionaler *Charaktertreiber*, der es mit speziellen Steuerbefehlen ermöglicht, den Cursor auf dem Bildschirm zu bewegen oder die Tastatur Ihres Zenith PC neu zu definieren. Damit ist es möglich, die standardisierten ANSI-Sequenzen aufzurufen. Die Datei wird durch den Befehl DEVICE = in der Datei CONFIG.SYS als Erweiterung des Betriebssystemes nachgeladen. Sie erweitert damit die Datei MSDOS.SYS. Ein normaler Aufruf als auszuführendes Programm ist nicht möglich.

Installation

Da es sich bei der Datei ANSI.SYS um einen Gerätetreiber (Device driver) handelt, der den schon vorhandenen Treiber mit dem Namen CON (s. a. Kapitel 8) unterstützt, muß diese Datei während des Systemstartes installiert werden. Dies geschieht mit Hilfe der Datei CONFIG.SYS. Fügen Sie in der Datei folgenden Befehl ein

```
DEVICE = ANSI.SYS
```

und booten Sie neu.

Bemerkungen

Der Gerätetreiber ANSI.SYS verhält sich wie der normale Treiber für die Eingabe von der Tastatur und Ausgabe auf dem Bildschirm mit dem Namen CON. Seine Besonderheit besteht darin, daß er die standardisierten ANSI-Steuersequenzen in maschinenspezifische Befehle umsetzen kann. Dadurch ist es möglich, Programme maschinenunabhängig zu schreiben. Differenzen der Hardware werden durch den Treiber ANSI.SYS ausgeglichen.

Die besonderen *ANSI-Sequenzen* werden von MS-DOS nur erkannt, wenn sie durch die Funktionsaufrufe 1, 2, 6 oder 9 weitergeleitet werden. Damit ist es dann möglich, den Cursor direkt zu positionieren, den Bildschirmmodus zu ändern oder die Tastatur neu zu belegen. Die einzelnen Funktionen werden in den folgenden Tabellen beschrieben. Sämtliche Funktionen werden mit dem Wert 27 dezimal oder 1B hexadezimal eingeleitet. Dies wird durch die Zeichenfolge ESC dargestellt. Das Zeichen # ist eine Variable, die mit dem entsprechenden dezimalen Wert einzugeben ist.

Cursorsteuerung

CUP-Cursorposition

ESC [#;#H

Der Befehl CUP bewegt den Cursor zu der durch die Variablen bestimmten Stelle auf dem Bildschirm. Die erste Variable bezeichnet die Zeile, die zweite die Spalte. Der vorgegebene Wert ist eins. Werden die Variablen nicht bestimmt, wird der Cursor in die linke obere Ecke gesetzt (Home-Position).

CUU - Cursor hoch

ESC [#A

Der Befehl CUU bewegt den Cursor eine Zeile nach oben, ohne die Spalte zu verändern. Der vorgegebene Wert ist eins. Der Befehl wird ignoriert, wenn der Cursor bereits in der obersten Zeile ist.

CUD - Cursor tief

ESC [#B

Der Befehl CUD bewegt den Cursor eine Zeile nach unten, ohne die Spalte zu verändern. Der vorgegebene Wert ist eins. Der Befehl wird ignoriert, wenn der Cursor bereits in der untersten Zeile ist.

CUF - Cursor rechts

ESC [#C

Der Befehl CUF bewegt den Cursor eine Spalte nach rechts, ohne die Zeile zu verändern. Der vorgegebene Wert ist eins. Der Befehl wird ignoriert, wenn der Cursor bereits in der äußersten rechten Spalte ist.

CUB - Cursor links

ESC [#D

Der Befehl CUD bewegt den Cursor eine Spalte nach links, ohne die Zeile zu verändern. Der vorgegebene Wert ist eins. Der Befehl wird ignoriert, wenn der Cursor bereits in der äußersten linken Spalte ist.

HVP - Cursorposition, horizontal und vertikal

ESC [#;#f

Der Befehl HVP bewegt den Cursor zu der durch die Variablen bestimmten Stelle auf dem Bildschirm. Die erste Variable bezeichnet die Zeile, die zweite die Spalte. Der vorgegebene Wert ist eins. Werden die Variablen nich bestimmt, wird der Cursor in die linke obere Ecke gesetzt (Home Position). Dieser Befehl entspricht CUP.

DSR - Cursorpositionsabfrage

ESC [6n

Der Befehl DSR fragt die momentane Cursorposition ab. Die Ausgabe erfolgt mit dem Befehl CPR (Cursorpositionsangabe).

CPR - Cursorpositionsangabe

ESC [#;#R

Dieser Befehl zeigt die momentane Cursorposition an. Die erste Variable bezeichnet die Zeile, die zweite die Spalte.

SCP - Cursorposition speichern

ESC [s

Der Befehl SCP speichert die momentane Cursorposition. Die Werte können mit dem Befehl RCP wieder abgefragt werden.

RCP - Cursorposition abfragen

ESC [u

Der Befehl RCP setzt den Cursor an die Stelle zurück, an der er war, als der Befehl SCP eingegeben wurde.

Löschen

ED - Bildschirm löschen

ESC [2J

Der Befehl ED löscht den gesamten Bildschirm und setzt den Cursor in die linke obere Ecke (entspricht CLS).

EL - Zeile löschen

ESC [K

Der Befehl EL löscht, beginnend ab der momentanen Cursorposition, die gesamte restliche Zeile.

Bildschirmsteuerung

SGR - Farbwahl

ESC [#;...;#m

Der Befehl SGR verändert die Farbe der darzustellenden Zeichen. Die Variablen bestimmen die neue Farbe. Nach Eingabe des Befehls SGR werden alle folgenden Zeichen in der neuen Farbe ausgegeben. Erst ein erneuter Befehl SGR verändert die Farbe wieder. Die Tabelle 9.4 zeigt eine komplette Übersicht der Farbmöglichkeiten.

Tabelle 9.4 Farbmöglichkeiten

WERT	FARBE
0	normal (schwarz-weiß)
1	intensiv
4	unterstreichen (nur für monochrome)
5	blinkend
7	invers (weiß-schwarz)
8	unsichtbar
30	schwarzer Vordergrund
31	roter Vordergrund
32	grüner Vordergrund
33	brauner Vordergrund
34	blauer Vordergrund
35	lila Vordergrund
36	blauer Vordergrund
37	weißer Vordergrund
40	schwarzer Hintergrund
41	roter Hintergrund
42	grüner Hintergrund
43	brauner Hintergrund
44	blauer Hintergrund
45	lila Hintergrund
46	blauer Hintergrund
47	weißer Hintergrund

Modifizieren der Datei MSDOS.SYS

SM - Modus auswählen

ESC [=#h
ESC [=h
ESC [=0h
ESC [?7h

Der Befehl SM verändert den Bildschirmmodus. Die Tabelle 9.5 zeigt alle Möglichkeiten.

Tabelle 9.5 Bildschirmmodus

MODUS	WERT	BESCHREIBUNG
Text	0	40 Zeichen/Zeile, 25 Zeilen/Schirm monochrome
Text	1	40 Zeichen/Zeile, 25 Zeilen/Schirm Farbe
Text	2	80 Zeichen/Zeile, 25 Zeilen/Schirm monochrome
Text	3	80 Zeichen/Zeile, 25 Zeilen/Schirm Farbe
Grafik	4	Mittlere Auflösung in Farbe, Textdarstellung mit 40 Zeichen/Zeile
Grafik	5	Mittlere Auflösung monochrome, Textdarstellung mit 40 Zeichen/Zeile
Grafik	6	Hochauflösende Grafik, monochrome, Textdarstellung mit 80 Zeichen/Zeile
	7	Zeichen nach Zeilenende werden in die nächste Zeile geschrieben

RM - Modus zurücksetzen

ESC [=#l
ESC [=l
ESC [=0l
ESC [?7l

Der Befehl RM entspricht dem Befehl SM. Es werden alle Funktionen ausgeführt, die in Tabelle 9.3 beschrieben sind. Bei Eingabe einer 7 werden Zeichen nach dem Zeilenende unterdrückt.

Neubelegung der Tastatur

Sie können die Tastatur Ihres Zenith PC mit Hilfe der ANSI-Steuerzeichen neu belegen. Die Steuerzeichen werden mit dem Befehl PROMPT (s. a. Kapitel 11) eingegeben. Sollen mehrere Tasten neu belegt werden, geschieht dies am einfachsten mit einer Batchdatei. Es ist nur darauf zu achten, daß maximal 250 neue Zeichen in der Datei ANSI.SYS dafür gespeichert werden können. Geben Sie mehr Zeichen ein, kann dies zu unvorhersehbaren Ergebnissen führen. Die Eingabe hierfür lautet:

```
ESC [#;#...#p
ESC ["Zeichenkette";p
ESC [#;"Zeichenkette";#;#"Zeichenkette";#p
```

Außerdem ist jede andere Kombination von numerischen Werten in dezimal und Zeichenkette möglich. Der erste Wert der Eingabe bezeichnet die Taste. Der nächste Wert bestimmt den neuen Wert der Taste. Ist der erste Wert gleich null, bestimmen der erste und zweite Wert die Taste (erweiterter ASCII-Kode). Soll beispielsweise die Taste Q ein A wiedergeben und umgekehrt, geben Sie folgende Befehle ein:

```
ESC [65;81p        (A wird Q)
ESC [97;113p       (a wird q)
ESC [81;65p        (Q wird A)
ESC [113;97p       (q wird a)
```

Ein weiteres Beispiel:

Der Befehl

```
ESC [0;68;"CLS";13p
```

belegt die Taste **F10** mit dem Befehl CLS und einem RETURN. Mit jedem Betätigen von F10 wird dann der gesamte Bildschirm gelöscht. Die beiden ersten Werte 0 und 68 bezeichnen die Taste **(F10)**, die Zeichenkette CLS ist das Befehlswort, und der Wert 13 bestimmt das RETURN.

VDISK.SYS

Die Datei VDISK.SYS ist ein funktionaler *Blocktreiber,* der es ermöglicht, im Arbeitsspeicher ein Diskettenlaufwerk zu simulieren. Dieses Laufwerk kann von Ihnen wie jedes andere normale Disketten- oder Winchesterlaufwerk benutzt werden. Es hat den Vorteil, erheblich schneller zu arbeiten. Sämtliche Operationen werden nur im Arbeitsspeicher ausgeführt. Der Zugriff auf die vergleichsweise langsamen externen Peripheriegeräte entfällt. Die gespeicherten Daten sind allerdings mit jedem Neustart oder nach dem Ausschalten des Gerätes verloren. Die Datei wird durch den Befehl DEVICE = in der Datei CONFIG.SYS als Erweiterung des Betriebssystemes nachgeladen. Sie erweitert damit die Datei MSDOS.SYS. Ein normaler Aufruf als auszuführendes Programm ist nicht möglich.

Installation

Da es sich bei der Datei VDISK.SYS um einen Gerätetreiber (Device driver) handelt, muß diese Datei während des Systemstartes installiert werden. Dies geschieht mit Hilfe der Datei CONFIG.SYS. Fügen Sie in der Datei folgenden Befehl ein:

```
DEVICE = [d:] [Weg] VDISK.SYS [B [S [D [/E]]]]
```

dabei bedeutet:

d:	vor dem Programmnamen bestimmt das aufzurufende Laufwerk
Weg	vor dem Programmnamen bestimmt das aufzurufende Verzeichnis
B	Größe des Laufwerkes in Kilobytes. Ohne Eingabe eines Wertes sind 64 KBytes vorgegeben.
S	Größe eines Sektors in Bytes. Ohne Eingabe sind 128 Bytes vorgegeben.
D	Anzahl der maximalen Einträge im Inhaltsverzeichnis. Ohne Eingabe sind 64 Einträge möglich.
/E	Das Laufwerk wird im Arbeitsspeicher oberhalb von 1 MByte eingerichtet. Dies ist nur möglich, wenn Ihr Zenith PC mit einer entsprechenden Speichererweiterung ausgerüstet ist.

Bemerkungen

In der *Befehlszeile* müssen nicht unbedingt Zusatzeingaben vorgenommen werden. Fehlen diese Eingaben, wird die Größe mit 64 KB festgesetzt und das Laufwerk ab der niedrigsten freien Adresse im Speicher installiert. Falls Sie Werte eingeben, achten Sie darauf, daß Sie die Angaben immer durch ein Leerfeld trennen. Bei jeder anderen Schreibweise werden die Zusatzeingaben nicht erkannt und die vorgegebenen Werte angenommen.

Den Zusatz /E können Sie nur beim Z-200 benutzen, falls Sie zusätzliche Speichererweiterungskarten eingebaut haben. Durch die Datei VDISK.SYS wird der Systemspeicher um ca. 720 Bytes vergrößert.

Rufen Sie den Treiber mehrmals auf, wird jedesmal ein weiteres Laufwerk eingerichtet. Mit dem Programm LABEL kann jedes Laufwerk einen eigenen Namen bekommen.

Laufwerkgröße

Die mit dem Zusatz B bestimmte Größe legt nur die ungefähre Größe der RAM-Disk fest. Wie bei jedem anderen Laufwerk auch benötigt das simulierte Laufwerk Speicherplatz zur internen Verwaltung (FAT und Inhaltsverzeichnis). Sie können jede beliebige Größe zwischen 1 KByte und dem maximalen Speicherausbau bestimmen. Der Treiber achtet jedoch darauf, daß noch ein freier Arbeitsspeicher von 64 KBytes zur Verfügung bleibt. Wählen Sie einen falschen Wert, wird die Größe automatisch auf 64 KBytes korrigiert.

Sektorgröße

Mit dieser Zusatzeingabe können Sie Platz sparen. Da MS-DOS Dateien immer nur sektorweise speichert, werden bei kleinen Dateien große Sektoren nur teilweise benutzt. Sie können mit diesem Zusatz eine Sektorgröße von 128, 256 oder 512 Bytes bestimmen. Geben Sie den Zusatz falsch ein, wird der Wert auf 128 Bytes korrigiert. Ebenso erfolgt eine Korrektur, falls bei zu kleiner Sektorgröße und zu großer Gesamtkapazität das FAT zu umfangreich würde.

Modifizieren der Datei MSDOS.SYS

Verzeichnisgröße

Mit diesem Zusatz bestimmen Sie die maximale Zahl der Einträge im Inhaltsverzeichnis. Jede Datei des Speicherlaufwerkes benötigt einen Eintrag. Ohne Eingabe des Zusatzes wird Platz für 64 Einträge reserviert. Bestimmen Sie eine größere Zahl, ohne diese jedoch zu nutzen, reservieren Sie unnötig Platz.

Sie können eine Zahl zwischen 2 und 512 eingeben. Je nach bestimmter Sektorgröße wird die Zahl entsprechend aufgerundet, da das Inhaltsverzeichnis immer nur ganze Sektoren belegt. Bestimmen Sie z. B. bei einer Sektorgröße von 256 Bytes 12 Einträge, errechnet VDISK daraus 16 Einträge.

Speichererweiterung

Haben Sie in Ihrem Zenith PC zusätzliche Speichererweiterungskarten installiert, können Sie das Laufwerk oberhalb von 1MByte einrichten. Sie bestimmen dies durch den Zusatz /E. Bei allen anderen Geräten wird der Zusatz ignoriert.

Zugriff

Der Zugriff auf das Speicherlaufwerk entspricht dem Zugriff aller anderen Laufwerke; es wird mit seinem Namen aufgerufen.

Das Betriebssystem MS-DOS bezeichnet alle angeschlossenen Laufwerke mit den Buchstaben des Alphabetes. Beginnend mit dem ersten Diskettenlaufwerk, das mit **A** bezeichnet wird, werden alle folgenden Laufwerke mit den nächsten Buchstaben benannt. Das Speicherlaufwerk wird zum Schluß installiert und daher mit dem letzten gültigen Buchstaben benannt. Verfügt Ihr Zenith PC nicht über ein Winchesterlaufwerk, wird das *Speicherlaufwerk* mit **C** bezeichnet. Die Buchstaben **A** und **B** sind immer für die ersten beiden Diskettenlaufwerke reserviert. Dies gilt auch für Geräte mit nur einem Laufwerk, da dies immer wie ein Gerät mit zwei Laufwerken behandelt wird (s. a. Kapitel 5). Ist Ihr Gerät mit einem Winchesterlaufwerk ausgerüstet, wird die *RAM-Disk* mit **G** bezeichnet, bei zwei Disketten- und einem Winchesterlaufwerk wird die *RAM-Disk* mit **I** bezeichnet.

Modifizieren der Datei MSDOS.SYS

Während der Installation werden die schon angeschlossenen Laufwerke abgefragt. Danach wird der RAM-Disk ein Name zugeteilt. Es erscheint die Meldung:

```
VDISK Version 3.02 virtual disk d:
Buffer size:         64 KB
Sector size:         128
Directory entries:   64
```

Dabei bezeichnet *d:* den gültigen Laufwerknamen, mit dem auf das Laufwerk zugegriffen werden kann. Zusätzlich werden alle weiteren Parameter gezeigt.

Benutzen Sie die Datei VDISK.SYS, um ein Laufwerk im Arbeitsspeicher zu simulieren, beachten Sie, daß es sich nur um ein vorübergehendes Laufwerk handelt. Sämtliche dort gespeicherten Dateien werden bei einem Neustart oder Ausschalten des Gerätes gelöscht. Bei jeder Installation wird das Laufwerk neu formatiert. Arbeiten Sie mit diesem Laufwerk, sollten Sie also in regelmäßigen Abständen die dort befindlichen Dateien kopieren.

Fehlermeldungen

Bad or missing VDISK.SYS	Datei VDISK.SYS fehlt oder ist fehlerhaft

ERKLÄRUNG:
Das Laufwerk kann nicht in der angegebenen Art installiert werden. Entweder ist die Datei VDISK.SYS nicht am bezeichneten Platz gespeichert, oder in der Eingabezeile wurden falsche Werte eingegeben (dies ist die häufigste Ursache für diese Fehlermeldung). Es ist auch möglich, daß der Arbeitsspeicher für die gewünschte Größe zu klein ist. Korrigieren Sie die Eingabezeile in der Datei CONFIG.SYS.

Invalid switch character	Falsche Zusatzeingabe

ERKLÄRUNG:
Diese Meldung erscheint, wenn Sie einen falschen Zusatz eingeben. Es ist nur /E erlaubt, andernfalls installiert das Betriebssystem das Laufwerk im niedrigen Arbeitsspeicher.

Modifizieren der Datei MSDOS.SYS

Beispiel - Datei

Die Datei CONFIG.SYS könnte folgenden Inhalt haben:

DEVICE = \SYSTEM\ANSI.SYS
DEVICE = \SYSTEM\VDISK.SYS 360 512 112
BREAK = ON
BUFFERS = 6
FILES = 12
COUNTRY = 49
LASTDRIVE = L

Zuerst werden die Dateien ANSI.SYS (für erweiterte Bildschirm- und Tastaturfunktionen) und VDISK.SYS (RAM-Disk) als Teil von MSDOS.SYS nachgeladen. Das Speicherlaufwerk hat eine Kapazität von 360 KB und entspricht damit einem normalen Diskettenlaufwerk. Es wird im niedrigen Speicherbereich installiert.

Der Befehl BREAK = ON modifiziert das Betriebssystem so, daß die Tastenkombination **CTRL-BREAK** das Programm jederzeit unterbricht.

Zusätzlich sind 6 Zwischenspeicher (insgesamt also 8) für Plattenzugriffe reserviert. Es können maximal 12 Dateien gleichzeitig geöffnet werden (fünf für das Betriebssystem und sieben für eigene Programme). Dann wird der Datensatz für Deutschland geladen und ermöglicht damit die Datumseingabe in der Form: Tag-Monat-Jahr. Die Zeit wird im 24-Stunden-Format angezeigt. Die Zahl der logischen Laufwerke wird auf 12 erhöht.

Fehlermeldungen

Bad or missing Command Interpreter	Befehlsinterpreter fehlt oder ist fehlerhaft

ERKLÄRUNG:
Entweder die Datei COMMAND.COM oder die mit dem Befehl SHELL bezeichnete Datei fehlt. Booten Sie mit einer anderen Diskette.

Configuration too large for memory	Modifiziertes System für den Speicher zu groß

ERKLÄRUNG:
Sie haben zu viele Erweiterungen der Datei MSDOS.SYS geladen. Es ist kein Platz mehr für den Befehlsinterpreter vorhanden. Booten Sie mit einer anderen Diskette.

Invalid Country Code	Falsche Landeskennziffer benannt

ERKLÄRUNG:
Sie haben mit dem Befehl COUNTRY eine falsche Kennziffer eingegeben. Die Datei wird dann weiter abgearbeitet, ohne diesen Befehl zu beachten. Ist der Bootvorgang abgeschlossen, korrigieren Sie den Fehler und booten neu.

Sector size too large in file	Sektor zu groß in Datei

ERKLÄRUNG:
Der bezeichnete Gerätetreiber ist fehlerhaft und wird nicht beachtet. Ist der Bootvorgang abgeschlossen, kopieren Sie eine fehlerfreie Fassung der Datei auf das Startlaufwerk und booten neu.

Too many Block Devices	Zu viele Gerätetreiber bestimmt

ERKLÄRUNG:
Sie haben mit dem Befehl DEVICE zu viele Gerätetreiber benannt. Der maximale Wert ist 26.

Fehlermeldungen

Unrecognized command in CONFIG.SYS	Nicht erkannter Befehl in der Datei CONFIG.SYS

Die Datei wird dann weiter abgearbeitet, ohne diesen Befehl zu beachten. Ist der Bootvorgang abgeschlossen, korrigieren Sie den Fehler und booten neu.

Zusammenfassung

Die meisten Betriebssysteme für Mikrocomputer haben folgende fünf Hauptbestandteile:

- Boot-Loader — lädt die restlichen Systemdateien in den Arbeitsspeicher, wird danach nicht mehr benötigt
- Ein-/Ausgabe Verwaltung — steuert die Ein- und Ausgabe
- Datei-verwaltung — steuert das Dateisystem
- Speicher-verwaltung — steuert den Arbeitsspeicher
- ausführendes Teil — lädt Programme und veranlaßt den Prozessor, die dort gespeicherten Anweisungen abzuarbeiten.

Um diese fünf Funktionen auszuführen, besteht das MS-DOS Betriebssystem aus den vier Teilen:

- BOOT-LOADER
- IO.SYS
- MSDOS.SYS
- COMMAND.COM

Die Teile IO.SYS und MSDOS.SYS können auf eine neue Diskette mit den Befehlen FORMAT und SYS kopiert werden. Es sind verdeckte Dateien. Sie können nicht verändert werden. Den Teil COMMAND.COM können Sie mit dem Befehl FORMAT kopieren, aber nicht mit dem Befehl SYS. Diese Datei ist nicht verdeckt. Der Teil BOOT-LOADER wird automatisch mit dem Befehl FORMAT auf den ersten Sektor (absolute Adresse: Seite 0, Spur 0, Sektor 1) jeder Diskette kopiert.

Das Betriebssystem MS-DOS wird in verschiedenen Schritten je nach Plattenart in den Computer geladen.

Zusammenfassung

Bootvorgang der Diskette

Zuerst lädt das Monitorprogramm den BOOT-LOADER. Der BOOT-LOADER lädt die Datei IO.SYS. Diese wiederum lädt die Datei MSDOS.SYS. Danach verschiebt das Modul SYSINIT (ein Teil von IO.SYS) die Datei MSDOS.SYS im Speicher und lädt anschließend die Datei CONFIG.SYS (falls vorhanden). Danach wird die Datei COMMAND.COM oder ein anderer Befehlsinterpreter geladen. Die Datei COMMAND.COM sucht die Datei AUTOEXEC.BAT und arbeitet (falls vorhanden) deren Befehle ab. Sonst erscheinen die Anforderungen für Datum und Zeit.

Bootvorgang der Winchester

Zuerst lädt das Monitorprogramm den Boot-Record. Der Boot-Record hat das Startlaufwerk gespeichert und lädt von diesem den BOOT-LOADER. Dieser lädt die Datei IO.SYS. Diese wiederum lädt die Datei MSDOS.SYS. Danach verschiebt das Modul SYSINIT (ein Teil von IO.SYS) die Datei MSDOS.SYS im Speicher und lädt anschließend die Datei CONFIG.SYS (falls vorhanden). Danach wird die Datei COMMAND.COM oder ein anderer Befehlsinterpreter geladen. Die Datei COMMAND.COM sucht die Datei AUTOEXEC.BAT und arbeitet (falls vorhanden) deren Befehle ab. Sonst erscheinen die Anforderungen für Datum und Zeit.

Bei der Eingabe eines Dateinamens, der zu mehr als einer Datei paßt, wird immer die Datei mit der höchsten Priorität geladen. Die Datei COMMAND.COM entscheidet immer in der Reihenfolge:

- interner Befehl
- externer Befehl mit dem Namenszusatz .COM
- externer Befehl mit dem Namenszusatz .EXE
- externer Befehl mit dem Namenszusatz .BAT

Die Datei MSDOS.SYS kann nach Ihren speziellen Wünschen erweitert werden. Dies geschieht mit Hilfe der Textdatei CONFIG.SYS.

Teil III

Befehlserklärungen

Befehlserklärungen

Dieser Teil des Handbuches erklärt ausführlich jeden einzelnen Befehl.

Das in diesem Teil vermittelte Wissen ist nicht unbedingt für jeden Benutzer des Zenith PC erforderlich. Dennoch sollten auch Benutzer, die nur Anwenderprogramme verwenden wollen, diesen Teil des Handbuches lesen. Sie können hier Hinweise erhalten, die es Ihnen ermöglichen, Ihren Computer und die mitgelieferten Programme voll zu nutzen.

Die Erklärungen erfolgen mit allgemeinverständlichen Worten, die auch von Lesern verstanden werden, die noch keine Erfahrung mit Mikrocomputern haben. Dabei werden nur die Begriffe vorausgesetzt, die in den ersten Kapiteln dieses Handbuches erklärt wurden.

Dieser Teil des Handbuches besteht aus den zwei Kapiteln:

- Kapitel 10 Befehlsübersicht
- Kapitel 11 Befehlserklärungen

Die hier nicht erklärten Befehle und Dienstprogramme werden entweder für die Herstellung von Computerprogrammen oder ausschließlich für Winchesterlaufwerke benötigt.

Erklärungen für diese Programme finden Sie im Teil IV „Weitere Dienstprogramme“ und im Teil V „Dienstprogramme für Winchesterlaufwerke“.

Kapitel 10

Befehlsübersicht

Übersicht

In diesem Kapitel werden sämtliche Befehle in mehreren Tabellen zusammengestellt. Ausführlich werden die Befehle dann in Kapitel 11 erklärt. Die Befehle werden zuerst alphabetisch aufgelistet (so wie sie in Kapitel 11 erklärt werden) und dann nach Funktionen geordnet. Dabei werden für die Eingaben unterschiedliche Variable benutzt:

D:	Laufwerkbezeichnung
p	Teil eines Winchesterlaufwerkes
S:	Ursprungslaufwerk
u:	Nummer des Laufwerkes
/X	Zusatzeingabe

Alphabetische Befehlsliste

Die Tabelle 10.1 enthält die alphabetische Liste der MS-DOS Befehle. Es wird außer dem Befehlsnamen die Art des Befehls angezeigt (I = Intern; E = Extern) sowie die Funktion und die richtige Eingabeform.

Befehlsübersicht

Alphabetische Befehlsliste

Tabelle 10.1 Alphabetische Befehlsliste

BEFEHL	ART	FUNKTION	EINGABE
APPLY	E	führt einen Befehl mehrere Male aus	APPLY [D:] Dateiname "Befehl %" [/S] APPLY "Befehl %" [D:] Dateiname [/S] APPLY [-] "Befehl %" [/S] APPLY "Befehl %" [-] [/S]
ASGNPART	E	bezeichnet einen Teil der Winchester als logisches Laufwerk	ASGNPART [?] ASGNPART u: ASGNPART u:p D:
ASSIGN	E	leitet Laufwerk-anfragen um	ASSIGN ? ASSIGN [x=y[...]]
ATTRIB	E	zeigt und ändert den Nur-Lese-Status von Dateien	ATTRIB Dateibez ATTRIB +R Dateibez ATTRIB -R Dateibez
BACKUP	E	sichert Dateien von Diskette oder Winchester	BACKUP BACKUP ? BACKUP [Dateibez[+Dateibez...] D: [/x...]]
BOOTF	E	schaltet das/die Winchesterlaufwerke ab	BOOTF
BREAK	I	überprüft auf Eingabe von CTRL-C oder CTRL-BREAK	BREAK [ON¦OFF]
CHDIR (CD)	I	verändert oder zeigt den Namen eines Inhalts-verzeichnisses	CHDIR [D:] [Weg] CD [D:] [Weg]

Alphabetische Befehlsliste

Tabelle 10.1 Alphabetische Befehlsliste

BEFEHL	ART	FUNKTION	EINGABE
CHKDSK	E	überprüft die Platte und/oder eine Datei auf Fehler	CHKDSK [D:] [Dateiname] [/X]
CIPHER	E	ver- und entschlüsselt Dateien	CIPHER Wort >Dateibez CIPHER Wort <Dateibez CIPHER Wort >Dateibez1>Dateibez2
CLS	I	löscht den Bildschirm	CLS
COMMAND	E	führt interne Befehle wie externe Befehle aus	COMMAND [D:] [Weg] [cttydev] [/X]
COMP	E	vergleicht den Inhalt zweier Dateien	COMP ? COMP Dateibez1 Dateibez2
CONFIGUR	E	modifiziert MS-DOS für Peripheriegeräte	CONFIGUR
COPY	I	kopiert Dateien	COPY Dateibez D: [/X] COPY Dateibez [D:] Dateiname [/X] COPY Dateibez [D:] Weg [/X] COPY [D:] Weg [D:] [Weg] [/X]
CTTY	I	wechselt zu einem anderen Eingabegerät	CTTY dev
DATE	I	zeigt und ändert das Tagesdatum	DATE DATE [mm-tt-jj]

Befehlsübersicht

Alphabetische Befehlsliste

Tabelle 10.1 Alphabetische Befehlsliste

BEFEHL	ART	FUNKTION	EINGABE
DEL (ERASE)	I	löscht Dateien	DEL Dateibez DEL [D:] Weg\Dateiname ERASE Dateibez ERASE [D:] Weg\Dateiname
DIR	I	zeigt Dateieinträge im Inhaltsverzeichnis	DIR [D:] [Dateiname] [/X] DIR [D:] [Weg] [/X]
DISKCOMP	E	vergleicht zwei Disketten	DISKCOMP [S: [D:]]
DISKCOPY	E	kopiert Disketten	DISKCOPY [S: [D:]] [/V]
ECHO	I	zeigt Batchdatei-befehle an	ECHO [ON¦OFF¦TEXT]
ERASE	I	siehe DEL	
EXIT	I	verläßt das Be-triebssystem und kehrt zum vorherigen Programm zurück	EXIT
FC	E	zeigt Unterschiede der Dateien	FC [/X] Dateinam1 Dateinam2
FIND	E	sucht einen Text	FIND [/X] "Zeichenkette" [Dateibez]
FOR	I	Befehlserweiterung	FOR %Variable IN (Wert) DO Befehl FOR %%Variable IN (Wert) DO Befehl
FORMAT	E	formatiert Diskette oder Winchester	FORMAT [D:] [/X...]
GOTO	I	überspringt Befehle	GOTO Bezeichnung

Alphabetische Befehlsliste

Tabelle 10.1 Alphabetische Befehlsliste

BEFEHL	ART	FUNKTION	EINGABE
GRAFTABL	E	lädt ASCII-Zeichen 128–255 für den Graphikmodus	GRAFTABL
IF	I	macht Befehlsausführung von einer Bedingung abhängig	IF [NOT] Bedingung Befehl
JOIN	E	ersetzt ein Laufwerk durch einen Weg	JOIN JOIN D: [D:] Weg JOIN D: /D
KEYBxxxx	E	lädt ein Tastaturanpassungsprogramm	KEYBxxxx
LABEL	E	speichert, ändert oder löscht einen Diskettennamen	LABEL [D:] [Name]
MKDIR (MD)	I	legt ein neues Inhaltsverzeichnis an	MKDIR [D:] Weg MD [D:] Weg
MODE	E	modifiziert MS-DOS für Peripheriegeräte	MODE ? MODE LPT#:[n][,[m][,P]] MODE [n][,[m][,T][[,s]] MODE COMn:Baud-Rate[,[Perity] [,[Data-Bits][,[Stop-Bits][,P]]]] MODE LPT#:=COMn
MORE	E	zeigt die Bildschirmausgabe seitenweise an	MORE < Dateibez Befehl ¦ MORE
NODEBUG	E	schaltet den Monitordebugger aus	NODEBUG

Befehlsübersicht

Alphabetische Befehlsliste

Tabelle 10.1 Alphabetische Befehlsliste

BEFEHL	ART	FUNKTION	EINGABE
PATH	I	benennt mehrere Verzeichnisse in Folge	PATH [D:] [Verzeichnis [;[D:] Verzeichnis[;..]
PAUSE	I	unterbricht die Befehlsabarbeitung	PAUSE [Bemerkung]
PRINT	E	druckt Dateien	PRINT PRINT Dateibez [/x] [...]
PROMPT	I	definiert das Anforderungszeichen	PROMPT [$] [Text]
PSCDrucker	E	gibt Grafikzeichen auf den Drucker	PSCDruckername
RDCPM	E	kopiert CP/M-Dateien	RDCPM RDCPM ? RDCPM DIR D: [Dateinam] [/Z] RDCPM [S:] Datei [D:] [Datei][/Z]
RECOVER	E	erneuert fehlerhafte Dateien	RECOVER D: RECOVER Dateibez
REM	I	zeigt einen Kommentar in einer Batchdatei	REM [Text]
RENAME (REN)	I	ändert den Dateinamen	REN Dateibez Dateiname RENAME Dateibez Dateiname
RESTORE	E	schreibt gesicherte Dateien zurück	RESTORE RESTORE ? RESTORE [[D:] [Dateibez[+Dateibez...]]] [/x...]
RMDIR (RD)	I	löscht ein Inhaltsverzeichnis	RMDIR [D:] Weg RD [D:] Weg

Tabelle 10.1 Alphabetische Befehlsliste

BEFEHL	ART	FUNKTION	EINGABE
SEARCH	E	sucht Dateien im gesamten Platten-verzeichnis	SEARCH [zdn] [/X]
SELECT	E	wählt Tastaturan-passung und Datums-format aus	SELECT yyy xxxx
SET	I	setzt eine Zeichen-kette einer anderen gleich	SET SET [Zeichenkette1=Zeichenkette2]
SHARE	E	reserviert Platz für File-Sharing	SHARE SHARE [/x . . .]
SHIFT	I	erlaubt mehr als 10 Variable in einer Batchdatei	SHIFT
SORT	E	sortiert, alpha-betisch oder numerisch	SORT [/X] [< Eingabe] [> Ausgabe]
SUBST	E	ersetzt ein Ver-zeichnis durch ein Laufwerk	SUBAST SUBST D: [D:] Weg SUBST D: /D
SYS	E	kopiert System-dateien	SYS D:
TIME	I	zeigt und ändert die Uhrzeit	TIME TIME hh[:mm[:ss[.cc]]]
TREE	E	zeigt das struk-turierte Verzeich-nis der Platte	TREE ? TREE [D:] [/F]

Befehlsübersicht

Alphabetische Befehlsliste

Tabelle 10.1 Alphabetische Befehlsliste

BEFEHL	ART	FUNKTION	EINGABE
TYPE	I	zeigt den Inhalt einer ASCII-Datei	TYPE Dateibez TYPE [D:] [Weg] Dateiname
VER	I	zeigt die Nummer der DOS Version	VER
VERIFY	I	prüft kopierte Dateien	VERIFY [ON¦OFF]
VOL	I	zeigt den Namen der Diskette oder Winchester	VOL [D:]

Funktionelle Befehlsliste

In diesem Abschnitt werden sämtliche Befehle noch einmal nach Funktionen zusammengestellt. Dabei ist es möglich, daß einige Befehle mehrmals aufgeführt werden, je nach ihrer Funktion.

Die folgenden Tabellen zeigen die Befehle außer mit dem Befehlsnamen, auch mit der Art des Befehls (I = Intern; E = Extern), der Funktion und der richtigen Eingabeform.

Analysierende Befehle

Die folgenden Befehle untersuchen, trennen und listen Ihre Daten und Programme auf.

Tabelle 10.2 Analysierende Befehle

BEFEHL	ART	FUNKTION	EINGABE
BACKUP	E	sichert Dateien von Diskette oder Winchester	BACKUP BACKUP ? BACKUP [Dateibez[+Dateibez...] D: [/x...]]
CHDIR (CD)	I	verändert oder zeigt den Namen eines Inhaltsverzeichnisses	CHDIR [D:] [Weg] CD [D:] [Weg]
CHKDSK	E	überprüft die Platte und/oder eine Datei auf Fehler	CHKDSK [D:] [Dateiname] [/X]
COMP	E	vergleicht den Inhalt zweier Dateien auf Gleichheit	COMP ? COMP Dateibez1 Dateibez2
DATE	I	zeigt und ändert das Tagesdatum	DATE DATE mm-tt-jj

Funktionelle Befehlsliste

Tabelle 10.2 Analysierende Befehle

BEFEHL	ART	FUNKTION	EINGABE
DIR	I	zeigt Dateieinträge im Inhaltsverzeichnis	DIR [D:] [Dateiname] [/X] DIR [D:] [Weg] [/X]
DISKCOMP	E	vergleicht zwei Disketten	DISKCOMP [S: [D:]]
FC	E	zeigt Unterschiede der Dateien	FC Dateinam1 Dateinam2 [/X]
LABEL	E	speichert, ändert oder löscht einen Diskettennamen	LABEL [D:] [Name]
RESTORE	E	schreibt gesicherte Dateien zurück	RESTORE RESTORE ? RESTORE [[D:] [Dateibez[+Dateibez...]]] [/x...]
TIME	I	zeigt und ändert die Uhrzeit	TIME TIME hh[:mm[:ss[.cc]]]
TREE	E	zeigt das strukturierte Verzeichnis der Platte	TREE ? TREE [D:] [/F]
TYPE	I	zeigt den Inhalt einer ASCII-Datei	TYPE Dateibez TYPE [D:] [Weg] Dateiname
VER	I	zeigt die Nummer der DOS Version	VER
VOL	I	zeigt den Namen der Diskette oder Winchester	VOL [D:]

Automatische Befehlsverarbeitung

Die folgenden Befehle sind hauptsächlich zur Verarbeitung in einer Batchdatei vorgesehen. Sie können aber auch direkt von Ihnen eingegeben werden.

Tabelle 10.3 Automatische Befehlsverarbeitung

BEFEHL	ART	FUNKTION	EINGABE
ECHO	I	zeigt Batchdatei-befehle an	ECHO [ON¦OFF¦Text]
FOR	I	Befehlserweiterung	FOR %Variable IN (Wert) DO Befehl FOR %%Variable IN (Wert) DO Befehl
GOTO	I	überspringt Befehle	GOTO Bezeichnung
IF	I	macht Befehlsausführung von einer Bedingung abhängig	IF [NOT] Bedingung Befehl
PAUSE	I	unterbricht die Befehlsabarbeitung	PAUSE [Bemerkung]
REM	I	zeigt einen Kommentar in einer Batchdatei	REM [Text]
SET	I	setzt eine Zeichenkette einer anderen gleich	SET SET [Zeichenkette1=Zeichenkette2]
SHIFT	I	erlaubt mehr als 10 Variable in einer Batchdatei	SHIFT

Funktionelle Befehlsliste

Befehle des Inhaltsverzeichnisses

Die folgenden Befehle werden zur Verwaltung der hierarchischen Struktur des Platteninhaltsverzeichnisses benötigt. Damit ist ein schneller Zugriff auf Dateien möglich.

Tabelle 10.4 Befehle des Inhaltsverzeichnisses

BEFEHL	ART	FUNKTION	EINGABE
CHDIR (CD)	I	verändert oder zeigt den Namen eines Inhaltsverzeichnisses	CHDIR [D:] [Weg] CD [D:] [Weg]
JOIN	E	ersetzt ein Laufwerk durch einen Weg	JOIN JOIN D: [D:] Weg JOIN D: /D
MKDIR (MD)	I	legt ein neues Inhaltsverzeichnis an	MKDIR [D:] Weg MD [D:] Weg
PATH	I	benennt mehrere Verzeichnisse in Folge	PATH [D:] [Verzeichnis [;[D:] Verzeichnis]...]
RMDIR (RD)	I	löscht ein Inhaltsverzeichnis	RMDIR [D:] Weg RD [D:] Weg
SEARCH	E	sucht Dateien im gesamten Plattenverzeichnis	SEARCH [zdn] [/X]
SUBST	E	ersetzt ein Verzeichnis durch ein Laufwerk	SUBST SUBST D: [D:] Weg SUBST D: /D
TREE	E	zeigt das strukturierte Verzeichnis der Platte	TREE ? TREE [D:] [/F]

Befehlsinterpreter

Dieser Befehl interpretiert eingegebene Befehle, überwacht deren korrekte Abarbeitung und ruft Programme auf.

Tabelle 10.5 Befehlsinterpreter

BEFEHL	ART	FUNKTION	EINGABE
COMMAND	E	ruft interne Befehle wie externe Befehle auf	COMMAND [D:] [Weg] [cttydev] [/X]

Befehlsübersicht

Funktionelle Befehlsliste

Dateiherstellung

Mit den folgenden Befehlen können Sie die Diskette oder Winchester zur Datenaufnahme vorbereiten, Inhaltsverzeichnisse erstellen und Dateien anlegen.

Tabelle 10.6 Dateiherstellung

BEFEHL	ART	FUNKTION	EINGABE
BACKUP	E	sichert Dateien von Diskette oder Winchester	BACKUP BACKUP ? BACKUP [Dateibez[+Dateibez...] D: [/x...]]
COPY	I	kopiert Dateien	COPY Dateibez D: [/X] COPY Dateibez [D:] Dateiname [/X] COPY Dateibez [D:] Weg [/X] COPY [D:] Weg [D:] [Weg] [/X]
DISKCOPY	E	kopiert Disketten	DISKCOPY [S: [D:]] [/V]
FORMAT	E	formatiert Diskette oder Winchester	FORMAT [D:] [/X...]
LABEL	E	speichert, ändert oder löscht einen Diskettennamen	LABEL [D:] [Name]
MKDIR (MD)	I	legt ein neues Inhaltsverzeichnis an	MKDIR [D:] Weg MD [D:] Weg
RDCPM	E	kopiert CP/M-Dateien	RDCPM RDCPM ? RDCPM DIR D: [Dateinam] [/Z] RDCPM [S:] Datei [D:] [Datei][/Z]
RESTORE	E	schreibt gesicherte Dateien zurück	RESTORE RESTORE ? RESTORE [[D:] [Dateibez[+Dateibez...]]] [/x...]
SYS	E	kopiert Systemdateien	SYS D:

Dateiveränderung

Mit den folgenden Befehlen können Sie Ihre Daten ändern, kopieren, löschen und beschädigte Dateien wiederherstellen.

Tabelle 10.7 Dateiveränderung

BEFEHL	ART	FUNKTION	EINGABE
APPLY	E	führt einen Befehl mehrere Male aus	APPLY [D:] Dateiname "Befehl %" [/S] APPLY "Befehl %" [D:] Dateiname [/S] APPLY [-] "Befehl %" [/S] APPLY "Befehl %" [-] [/S]
ATTRIB	E	zeigt und ändert den Nur-Lese-Status von Dateien	ATTRIB Dateibez ATTRIB +R Dateibez ATTRIB -R Dateibez
BACKUP	E	sichert Dateien von Diskette oder Winchester	BACKUP BACKUP ? BACKUP [Dateibez[+Dateibez...] D: [/x...]]
CIPHER	E	ver- und entschlüsselt Dateien	CIPHER Wort >Dateibez CIPHER Wort <Dateibez CIPHER Wort <Dateibez1>Dateibez2
COPY	I	kopiert Dateien	COPY Dateibez D: [/X] COPY Dateibez [D:] Dateiname [/X] COPY Dateibez [D:] Weg [/X] COPY [D:] Weg [D:] [Weg] [/X]
DEL (ERASE)	I	löscht Dateien	DEL Dateibez DEL [D:] Weg\Dateiname ERASE Dateibez ERASE [D:] Weg\Dateiname

Funktionelle Befehlsliste

Tabelle 10.7 Dateiveränderung

BEFEHL	ART	FUNKTION	EINGABE
DISKCOPY	E	kopiert Disketten	DISKCOPY [S: [D:]] [/V]
NODEBUG	E	schaltet den Monitor-debugger aus	NODEBUG
RDCPM	E	kopiert CP/M-Dateien	RDCPM RDCPM ? RDCPM DIR D: [Dateinam] [/Z] RDCPM [S:] Datei [D:] [Datei][/Z]
RECOVER	E	erneuert fehler-hafte Dateien	RECOVER D: RECOVER Dateibez
REN RENAME	I	ändert den Dateinamen	REN Dateibez Dateiname RENAME Dateibez Dateiname
RESTORE	E	schreibt gesicherte Dateien zurück	RESTORE RESTORE ? RESTORE [[D:] [Dateibez[+Dateibez...]]] [/x...]

Dateiverschiebung

Mit den folgenden Befehlen können Sie Daten zwischen den angeschlossenen Peripheriegeräten bewegen.

Tabelle 10.8 Dateiverschiebung

BEFEHL	ART	FUNKTION	EINGABE
BACKUP	E	sichert Dateien von Diskette oder Winchester	BACKUP BACKUP ? BACKUP [Dateibez[+Dateibez...] D: [/x...]]
COPY	I	kopiert Dateien	COPY Dateibez D: [/X] COPY Dateibez [D:] Dateiname [/X] COPY Dateibez [D:] Weg [/X] COPY [D:] Weg [D:] [Weg] [/X]
DISKCOPY	E	kopiert Disketten	DISKCOPY [S: [D:]] [/V]
FORMAT	E	formatiert Diskette oder Winchester	FORMAT [D:] [/x...] RECOVER D: RECOVER Dateibez
RECOVER	E	erneuert fehlerhafte Dateien	
RESTORE	E	schreibt gesicherte Dateien zurück	RESTORE RESTORE ? RESTORE [[D:] [Dateibez[+Dateibez...]]] [/x...]
SYS	E	kopiert Systemdateien	SYS D:

Funktionelle Befehlsliste

Druckbefehle

Mit den folgenden zwei Befehlen können Sie Daten aus ASCII-Dateien oder vom Bildschirm auf den mit PRN bezeichneten Drucker ausgeben.

Tabelle 10.9 Druckbefehle

BEFEHL	ART	FUNKTION	EINGABE
PRINT	E	druckt Dateien	PRINT PRINT Dateibez [/X] [Dateibez[/X].]
PSCDrucker	E	gibt Grafikzeichen auf den Drucker	PSCDruckername

Filterbefehle

Die folgenden Befehle lesen Daten von dem Standardeingabegerät (Konsole). Sie verändern die Daten und geben Sie zum Standardausgabegerät (Bildschirm). Filterbefehle werden häufig in Verbindung mit Befehlsverkettung benutzt.

Tabelle 10.10 Filterbefehle

BEFEHL	ART	FUNKTION	EINGABE
CIPHER	E	ver- und entschlüsselt Dateien	CIPHER Wort >Dateibez CIPHER Wort <Dateibez CIPHER Wort <Dateibez1>Dateibez2
FIND	E	sucht einen Text	FIND [/X] "Zeichenkette" [Dateibez]
MORE	E	zeigt die Bildschirmausgabe seitenweise an	MORE < Eingabe Befehl ¦ MORE
SORT	E	sortiert alphabetisch oder numerisch	SORT [/X]

Plattenveränderung

Mit den folgenden Befehlen können Sie den Status der Diskette oder Winchester und die Verwaltung der Daten beeinflussen.

Tabelle 10.11 Plattenveränderung

BEFEHL	ART	FUNKTION	EINGABE
ASGNPART	E	bezeichnet einen Teil der Winchester als logisches Laufwerk	ASGNPART [?] ASGNPART u: ASGNPART u:p D:
ASSIGN	E	leitet Laufwerk-anfragen um	ASSIGN ? ASSIGN [x=y[...]]
ATTRIB	E	zeigt und ändert den Nur-Lese-Status von Dateien	ATTRIB Dateibez ATTRIB +R Dateibez ATTRIB -R Dateibez
CHDIR (CD)	I	verändert oder zeigt den Namen eines Inhalts-verzeichnisses	CHDIR [D:] [Weg] CD [D:] [Weg]
DEL (ERASE)	I	löscht Dateien	DEL Dateibez DEL [D:] Weg Dateiname ERASE Dateibez ERASE [D:] Weg Dateiname
FORMAT	E	formatiert Diskette oder Winchester	FORMAT [D:] [/X...]

Tabelle 10.11 Plattenveränderung

BEFEHL	ART	FUNKTION	EINGABE
NODEBUG	E	schaltet den Monitordebugger aus	NODEBUG
PATH	I	benennt mehrere Verzeichnisse in Folge	PATH [D:] [Verzeichnis [;[D:] Verzeichnis]...]
REN RENAME	I	ändert den Dateinamen	REN Dateibez Dateiname RENAME Dateibez Dateiname
RMDIR (RD)	I	löscht ein Inhaltsverzeichnis	RMDIR [D:] Weg RD [D:] Weg

Funktionelle Befehlsliste

Sicherheitsbefehle

Mit den folgenden Befehlen können Sie Fehler lokalisieren oder beschädigte Daten erneuern.

Tabelle 10.12 Sicherheitsbefehle

BEFEHL	ART	FUNKTION	EINGABE
BREAK	I	überprüft auf Eingabe von CTRL-C oder CTRL-BREAK	BREAK [ON¦OFF]
CHKDSK	E	überprüft die Platte und/oder eine Datei auf Fehler	CHKDSK [D:] [Dateiname] [/X]
COMP	E	vergleicht den Inhalt zweier Dateien auf Gleichheit	COMP ? COMP Dateibez1 Dateibez2
DISKCOMP	E	vergleicht zwei Disketten	DISKCOMP [S: [D:]]
FC	E	zeigt Unterschiede der Dateien	FC [/X] Dateinam1 Dateinam2
RECOVER	E	erneuert fehlerhafte Dateien	RECOVER D: RECOVER Dateibez
VERIFY	I	prüft kopierte Dateien	VERIFY [ON] VERIFY [OFF]

Systemvorbereitungsbefehle

Mit den folgenden Befehlen bereiten Sie Daten und Geräte für die Benutzung vor.

Tabelle 10.13 Systemvorbereitungsbefehle

BEFEHL	ART	FUNKTION	EINGABE
ASSIGN	E	leitet Laufwerkanfragen um	ASSIGN ? ASSIGN [x=y[...]]
CONFIGUR	E	modifiziert MS-DOS für Peripheriegeräte	CONFIGUR
CTTY	I	wechselt zu einem anderen Eingabegerät	CTTY dev
FORMAT	E	formatiert Diskette oder Winchester	FORMAT [D:] [/X...]
MODE	E	modifiziert MS-DOS für Peripheriegeräte	MODE ? MODE LPT#:[n][,[m][,P]] MODE [n][,[m][,T]][,s]] MODE COMn:Baud-Rate[,[Parity] [,[Data-Bits][,[Stop-Bits][,P]]]] MODE LPT#:=COMn
NODEBUG	E	schaltet den Monitordebugger aus	NODEBUG
SET	I	setzt eine Zeichenkette einer anderen gleich	SET SET [Zeichenkette1=Zeichenkette2]
SYS	E	kopiert Systemdateien	SYS D:

Funktionelle Befehlsliste

Verschiedene Befehle

Die folgenden Befehle rufen verschiedene MS-DOS Funktionen auf. Sie sind für ein korrektes Arbeiten des Systems nicht unbedingt erforderlich.

Tabelle 10.14 Verschiedene Befehle

BEFEHL	ART	FUNKTION	EINGABE
ASSIGN	E	leitet Laufwerkanfragen um	ASSIGN ? ASSIGN [x=y[...]]
CLS	I	löscht den Bildschirm	CLS
DATE	I	zeigt und ändert das Tagesdatum	DATE DATE mm-tt-jj
EXIT	I	verläßt das Betriebssystem und kehrt zum vorherigen Programm zurück	EXIT
JOIN	E	ersetzt ein Laufwerk durch einen Weg	JOIN JOIN D: [D:] Weg JOIN D: /D
LABEL	E	speichert, ändert oder löscht einen Diskettennamen	LABEL [D:] [Name]
PROMPT	I	definiert das Anforderungszeichen	PROMPT [$] [Text]
REM	I	zeigt einen Kommentar in einer Batchdatei	REM [Text]

Tabelle 10.14 Verschiedene Befehle

BEFEHL	ART	FUNKTION	EINGABE
SET	I	setzt eine Zeichenkette einer anderen gleich	SET SET [Zeichenkette1=Zeichenkette2]
SUBST	E	ersetzt ein Verzeichnis durch ein Laufwerk	SUBST SUBST D: [D:] Weg SUBST D: /D
TIME	I	zeigt und ändert die Uhrzeit	TIME TIME hh[:mm[:ss[.cc]]]
TREE	E	zeigt das strukturierte Verzeichnis der Platte	TREE ? TREE [D:] [/F]
VER	I	zeigt die Nummer der DOS Version	VER
VOL	I	zeigt den Namen der Diskette oder Winchester	VOL [D:]

Kapitel 11

Befehlserklärungen

Übersicht

In diesem Kapitel werden sämtliche *Befehle* zur Handhabung des Betriebssystemes ausführlich beschrieben. Die Erklärungen erfolgen in einfachen Worten, die kein spezielles Wissen voraussetzen. Es ist lediglich erforderlich, daß Sie die in den ersten Kapiteln erklärten Begriffe kennen.

Die Erklärung der Befehle erfolgt in alphabetischer Reihenfolge. Es wird der Zweck und die Schreibweise des Befehls erklärt. Im Abschnitt Bemerkungen wird die Arbeitsweise ausführlich beschrieben. Zum Schluß eines jeden Befehls werden mögliche Fehlermeldungen besprochen.

APPLY Extern

Zweck

führt einen Befehl mehrere Male mit verschiedenen Dateien aus

Syntax

[d:] [Weg] APPLY ?
[d:] [Weg] APPLY [D:] Dateiname "Befehl %" [/S]
[d:] [Weg] APPLY [-] "Befehl %" [/S]
[d:] [Weg] APPLY "Befehl %" [D:] Dateiname [/S]
[d:] [Weg] APPLY "Befehl %" [-] [/S]

dabei bedeutet:

d:	vor dem Programmnamen bestimmt das aufzurufende Laufwerk
Weg	vor dem Programmnamen bestimmt das aufzurufende Verzeichnis
?	es erscheint eine Erklärung des Programmes
D:	Laufwerkbezeichnung der Datei, welche die Eingaben für APPLY enthält
Dateiname	Name der Datei, welche die Eingaben für APPLY enthält
–	Bezeichnung für die Standardeingabe
Befehl	Name des Befehls, der mehrmals ausgeführt werden soll
%	Variable (zu bearbeitende Datei)
/S	unterdrückt die Anzeige der Befehlszeile auf dem Bildschirm

Bemerkungen

Das Programm APPLY kann im *Netzwerk* **ohne** Einschränkungen benutzt werden.

Mit diesem Dienstprogramm können Sie einen *Befehl* mit verschiedenen Variablen (Dateien) *mehrmals* ausführen. Es ist nicht mehr notwendig, den Befehl immer wieder einzugeben. Voraussetzung für ein korrektes Arbeiten des Dienstprogrammes ist die Datei COMMAND.COM. Sie muß immer im Arbeitslaufwerk vorhanden sein.

Das Programm APPLY liest seine Eingaben von der Standardeingabe. Es ist aber auch möglich, dieses Programm mit anderen Befehlen zu verketten. Nähere Informationen zur Befehlsverkettung finden Sie im Kapitel 8.

Rufen Sie APPLY mit einer Datei zusammen auf, liest APPLY jede Zeile dieser Datei. Er sucht dabei nach einer *Variablen*, die er in der Befehlszeile für das Variablenzeichen (%) einsetzen kann. Er führt dann den bezeichneten Befehl einmal mit jeder Variablen aus. Dabei wird vor jeder Befehlsausführung die Befehlszeile auf dem Bildschirm angezeigt.

Sämtliche Eingaben in der Befehlszeile, mit der Sie das Programm APPLY aufrufen, müssen durch eine Leertaste getrennt werden. Ebenso ist es unbedingt erforderlich, daß Sie den auszuführenden Befehl benennen. Vom Programm APPLY kann nicht auf gespeicherte Werte zurückgegriffen werden.

Programmaufruf

Es gibt drei verschiedene Möglichkeiten, das Dienstprogramm aufzurufen und damit zu arbeiten:

- Sie geben den Namen einer Datei ein, die die Variablen gespeichert hat.
- Sie verketten das Programm mit einem anderen Befehl.
- Sie rufen das Programm auf und geben die Variable jedesmal über die Tastatur ein.

Haben Sie die Namen der zu bearbeitenden *Variablen* (meistens Dateinamen) in einer separaten Datei gespeichert, können Sie den Namen der Datei in der Befehlszeile mit eingeben. Das Programm APPLY ruft diese Datei auf und liest jede Zeile dieser Datei in eine neue Befehlszeile ein. Die neue Befehlszeile besteht aus dem Befehlsnamen mit dem Variablenzeichen (%).

Haben Sie z. B. in der Datei ALLBAT.ASC sämtliche Namen von Batchdateien gespeichert, können Sie alle Batchdateien mit dem Befehl TYPE anzeigen lassen. Rufen Sie das Programm APPLY dann folgendermaßen auf:

```
APPLY ALLBAT.ASC "TYPE %"
```

Jetzt wird vom Programm APPLY jede Zeile der Datei ALLBAT.ASC für das Prozentzeichen eingesetzt. Diese neue Befehlszeile wird zur Datei COMMAND.COM weitergegeben, die den so entstandenen Befehl ausführt. Die neue Befehlszeile lautet z. B.:

```
TYPE AUTOEXEC.BAT
```

Die Befehlszeile wird noch einmal angezeigt, danach wird der Befehl ausgeführt.

/S – ohne Befehlsanzeige

Durch Eingabe des Zusatzes /S kann die Ausgabe der Befehlszeile auf dem Bildschirm unterdrückt werden. Alle Meldungen der aufgerufenen Programme erscheinen weiterhin unverändert. Rufen Sie das Programm dann folgendermaßen auf:

```
APPLY ALLBAT.ASC "TYPE %" /S
```

APPLY

Beispiele

Verketten Sie das Programm APPLY mit einem anderen Befehl, nehmen Sie die Ausgabedaten dieses Befehls als Eingabedaten von APPLY. Ein Beispiel:

```
SEARCH C:*.* ¦SORT¦APPLY "COPY % B:"
```

Das Programm SEARCH erstellt eine Liste mit den Dateinamen des Laufwerkes C:. Diese Liste wird dem Filter SORT zum Sortieren übergeben, der sie dann zum Programm APPLY weiterreicht. Von APPLY werden die Dateien dann auf das Laufwerk B: kopiert. Auf dem Bildschirm wird laufend der jeweilige Arbeitsgang angezeigt, Zuerst wird die zu kopierende Datei genannt.

```
COPY ... B:
```

Ist die Datei kopiert, erscheint die Meldung:

```
1 File(s) copied                1 Datei kopiert
```

Wenn Sie das Programm APPLY nur mit dem auszuführenden Befehl aufrufen, können Sie den Namen der zu bearbeitenden Datei oder jede andere Variable jedesmal über die Tastatur eingeben. In diesem Fall liest das Programm APPLY die Standardeingabe. Das Programm kennt kein Befehlsanforderungszeichen (Prompt). Ein Beispiel:

```
APPLY "CHKDSK %:"
```

Geben Sie jetzt nur noch den Buchstaben für das Laufwerk ein (ohne Doppelpunkt!). Jedes Laufwerk kann dann vom Programm CHKDSK überprüft werden. Geben Sie z. B. ein A ein, erscheint in der Befehlszeile

```
CHKDSK A:
```

und anschließend die Statusmeldung für das Laufwerk A:.

Fehlermeldungen

Außer den Fehlermeldungen, die durch das auszuführende Programm hervorgerufen werden, könnten folgende Meldungen beim Aufruf des Programmes APPLY erscheinen:

Error EXECing COMMAND.COM, Terminating	Fehler durch Aufruf der Datei COMMAND.COM, Befehl abgebrochen

ERKLÄRUNG:
Nach dem Aufruf kann das Programm APPLY die Datei COMMAND.COM nicht finden. Kopieren Sie den Befehlsinterpreter in das Rootdir. Benennen Sie keinen Weg.

APPLY: Command string required	APPLY: Befehlseingabe erforderlich

Erklärung:
Sie haben beim Aufruf des Programmes APPLY nicht den auszuführenden Befehl mit eingegeben.

APPLY: More than one command string given	APPLY: Es wurden mehrere Befehle eingegeben

Erklärung:
Sie haben beim Aufruf des Programmes APPLY mehrere auszuführende Befehle eingegeben.

APPLY: More than one file name specified	APPLY: Es wurden mehrere Dateien benannt

ERKLÄRUNG:
Sie haben beim Aufruf des Programmes APPLY mehrere Eingabedateien mit Variablen benannt.

APPLY: Unable to locate requested input file	APPLY: Bezeichnete Eingabedatei nicht vorhanden

ERKLÄRUNG:
Das Programm APPLY kann die Datei mit den Variablen nicht im Inhaltsverzeichnis finden.

ASSIGN Extern

Zweck

leitet Laufwerkanfragen eines Anwenderprogrammes auf ein anderes Laufwerk um

Syntax

[d:] [Weg] ASSIGN
[d:] [Weg] ASSIGN ?
[d:] [Weg] ASSIGN [x=y [...]]

dabei bedeutet:

d:	vor dem Programmnamen bestimmt das aufzurufende Laufwerk
Weg	vor dem Programmnamen bestimmt das aufzurufende Verzeichnis
?	zeigt eine kurze Erklärung des Programmes
x	vom Programm bestimmtes Laufwerk
y	tatsächliches Laufwerk (mit ASSIGN neu bestimmt)

Bemerkungen

Das Programm ASSIGN kann im *Netzwerk* **nicht** benutzt werden.

Das Dienstprogramm ASSIGN ermöglicht es Ihnen, Laufwerkanfragen eines Anwenderprogrammes für ein bestimmtes Laufwerk (z. B. B:) vorübergehend auf ein anderes Laufwerk (z. B. C:) umzuleiten. Damit können Anwenderprogramme, die für ein Gerät mit zwei Diskettenlaufwerken konzipiert wurden, ohne Änderungen auf einem Gerät mit einem Winchesterlaufwerk eingesetzt werden. Dies ist immer dann gegeben, wenn das Anwenderprogramm die Programmdiskette im Laufwerk A: verlangt und die Datendiskette im Laufwerk B:.

Bietet das Anwenderprogramm nicht die Möglichkeit in einem Installationsmodul die zu benutzenden Laufwerke selber zu bestimmen (z. B. C: für Programme und Daten gleichzeitig), kann mit Hilfe des Dienstprogrammes ASSIGN diese Aufgabe dem Betriebssystem übertragen werden. Dies ist sowohl mit Disketten- als auch mit Winchesterlaufwerken oder mit Teilbereichen eines Winchesterlaufwerkes möglich.

ACHTUNG:
Benutzen Sie das Program ASSIGN nicht in Verbindung mit den Dienstprogrammen von MS-DOS. Durch die Umleitung können die wahren Eigenschaften eines Laufwerks verdeckt werden. Einige Programme benötigen spezifische Daten des zu bearbeitenden Laufwerkes. Werden diese Daten nicht richtig erkannt, können unvorhersehbare Reaktionen auftreten. Daher sollten Sie die folgenden Programme niemals für ein umbenanntes Laufwerk benutzen:

ASGNPART
BACKUP (mit Formatieren)
DISKCOMP
DISKCOPY
FORMAT
PRINT
RDCPM

Haben Sie für ein Anwenderprogramm Laufwerke vorübergehend umbenannt, sollten Sie sofort nach Beendigung des Programmes alle Umbenennungen rückgängig machen. Alle Umbenennungen werden vom Betriebssystem so lange gespeichert, bis sie durch einen weiteren ASSIGN-Befehl überschrieben werden, bis sie gelöscht werden, oder bis Sie das System neu starten.

Programmaufruf

Sie können sich jederzeit für das Programm ASSIGN eine kurze Erklärung anzeigen lassen. Geben Sie dafür den Befehl ein:

```
ASSIGN ?
```

ASSIGN

Es erscheint dann folgender Text:

ASSIGN Version 3.04
Copyright (C) 1985 Zenith Data Systems Corporation

ASSIGN instructs the operating system to use a different drive from that specified by an application program for disk operations. That is, ASSIGN is used to remap system drive name assignments. If entered without any parameters, ASSIGN resets normal system drive assignments.

Syntax: ASSIGN [x=y [...]]

Multiple drives can be remapped with one command; however, the ASSIGN command line should not exceed 127 characters in length. Use a space, comma, semicolon, or tab to separate pairs of drive names. Drive names may be entered in uppercase, lowercase, or both, and need not be followed ba a colon.

Examples:

The following command causes all references to drives A, B, and D to be redirected to drives C, C, and E respectively: ASSIGN A=C B=C D=E

The following command resets drive mappings to their original states (any previous remapping or redirection of drives invoked with ASSIGN is cancelled): ASSIGN

Benutzen Sie ein Programm, das z. B. immer im Laufwerk A: die Programmdiskette verlangt, können Sie mit folgendem Beispiel alle Zugriffe auf das Laufwerk C: umleiten:

ASSIGN A=C

Es wird nach dem Befehlswort (ASSIGN) erst der Name des umzuleitenden Laufwerkes (hier A) eingegeben, dann der Name des neuen Laufwerkes, auf das wirklich zurückgegriffen werden soll (hier C). Es ist nicht erforderlich, nach den Buchstaben für die Laufwerkbezeichnungen einen Doppelpunkt mit einzugeben. Sollen mit einem Befehl mehrere Umbenennungen zugleich vorgenommen werden, geben Sie, durch ein gültiges MS-DOS-Trennzeichen getrennt, weitere Laufwerkpaare ein. Ein Beispiel:

ASSIGN A=C B=C

Mit diesem Befehl wird das System angewiesen, sämtliche Zugriffe für die Laufwerke A: und B: auf das Laufwerk C: weiterzuleiten. Mit einem Befehl können beliebig viele Umbenennungen vorgenommen werden. Dies wird nur durch die Länge der Befehlszeile insgesamt mit 128 Zeichen begrenzt.

Sollen die Umbenennungen wieder gelöscht werden, geben Sie den Befehl ohne einen Zusatz ein. Der Befehl

```
ASSIGN
```

löscht alle gespeicherten Veränderungen. Jetzt wird jedes Laufwerk wieder mit seinem ursprünglichen Namen erreicht. Ebenso werden alle Veränderungen mit einem Neustart des Systemes gelöscht.

Fehlermeldungen

Incorrect DOS Version	Falsche DOS-Version

ERKLÄRUNG:
Sie benutzen das Programm ASSIGN unter einer DOS-Version, die nicht 3.0 oder höher ist.

Invalid parameter	Falsche Zusatzeingabe

ERKLÄRUNG:
Sie haben nach dem Befehlswort ASSIGN zu viele oder zu wenige Eingaben vorgenommen. Geben Sie den Befehl in der richtigen Schreibweise neu ein.

Cannot ASSIGN a Network drive	Ein Laufwerk des Netzwerkes kann nicht umgeleitet werden

ERKLÄRUNG:
Sie haben versucht, mit dem Befehl ASSIGN ein Laufwerk des Netzwerkes umzuleiten. Dies ist nicht möglich.

ATTRIB Extern

Zweck

zeigt und ändert den Nur-Lese-Status von Dateien

Syntax

[d:] [Weg] ATTRIB Dateibez
[d:] [Weg] ATTRIB +R Dateibez
[d:] [Weg] ATTRIB -R Dateibez

dabei bedeutet:

d:	vor dem Programmnamen bestimmt das aufzurufende Laufwerk
Weg	vor dem Programmnamen bestimmt das aufzurufende Verzeichnis
Dateibez	Name der Datei, die bearbeitet werden soll
+R	setzt das Nur-Lese-Bit
-R	löscht das Nur-Lese-Bit

Bemerkungen

Das Programm ATTRIB kann im *Netzwerk* **ohne** Einschränkungen benutzt werden.

MS-DOS speichert im Inhaltsverzeichnis der Platte zu jedem Dateinamen weitere Informationen über den Eintrag. Dies geschieht in einem besonderen Byte. Damit kann MS-DOS besondere Eigenschaften des Eintrages erkennen. Dazu gehört auch der Status einer *Nur-Lese-Datei.* Dies wird durch ein spezielles Bit gekennzeichnet. Nähere Informationen dazu finden Sie unter dem Befehl FORMAT ab Seite 11.151.

Das Programm ATTRIB verändert dieses Bit. Bei gesetztem Bit beschreibt oder löscht MS-DOS die Datei nicht mehr, Lesezugriffe werden wie gewohnt durchgeführt. Damit ist die Datei softwaremäßig schreibgeschützt. Arbeiten Sie im Netzwerk, können Sie damit Ihre Dateien leicht vor Veränderungen durch andere Benutzer schützen.

Rufen Sie das Programm nur mit einer Dateibezeichnung auf, sehen Sie den Status der gewünschten Datei. Vor dem Dateinamen darf eine Laufwerkbezeichnung und/oder ein Verzeichnisname eingegeben werden. Sollen alle Dateien eines anderen Verzeichnisses (nicht des momentanen Arbeitsverzeichnisses) aufgelistet werden, müssen Sie nach dem Verzeichnisnamen die Zeichen „*.*" eingeben.

Sie ändern den Status der Datei mit der Zusatzeingabe R. Geben Sie den Zusatz mit einem Pluszeichen (+) ein, setzen Sie das Nur-Lese-Bit der Datei und verhindern damit alle Schreibzugriffe. Geben Sie den Zusatz mit einem Minuszeichen (-) ein, löschen Sie das Nur-Lese-Bit und es sind wieder alle Dateizugriffe erlaubt. Rufen Sie einen Befehl zweimal für dieselbe Datei auf, wird er das zweite Mal ignoriert.

Beispiele

Die folgenden Beispiele zeigen Ihnen die drei möglichen Eingaben des Programmes.

1. Status abfragen

Um den Status einer oder mehrerer Dateien des momentanen Laufwerkes abzufragen, geben Sie z. B. ein:

```
ATTRIB *.COM
```

Es könnte folgende Liste erscheinen:

```
        A:\COMMAND.COM
R       A:\ASSIGN.COM
        A:\DEBUG.COM
R       A:\ASGNPART.COM
        A:\TREE.COM
        A:\COMP.COM
        A:\SEARCH.COM
R       A:\APPLY.COM
        A:\CIPHER.COM
```

Die Programme ASSIGN.COM, ASGNPART.COM und APPLY .COM sind mit einem Schreibschutz versehen.

2. Attribut setzen

Im folgenden Beispiel sollen alle Dateien mit dem Anfangsbuchstaben C ebenfalls mit einem Schreibschutz versehen werden. Geben Sie dafür ein:

ATTRIB +R *.COM

Das Ergebnis können Sie anschließend durch Eingabe des ersten Beispieles überprüfen.

3. Attribut löschen

Im letzten Beispiel soll das Attribut von allen Dateien wieder entfernt werden. Geben Sie dafür ein:

ATTRIB -R *.COM

Fehlermeldung

Invalid number of parameters	Falsche Zusatzeingaben

ERKLÄRUNG:
Sie haben das Programm ATTRIB ohne weitere Eingaben aufgerufen. Sie müssen mindestens einen Dateinamen eingeben.

BACKUP c:\basic*.* a:/n/s

BACKUP Extern

Zweck

sichert Dateien von Diskette oder Winchester

Syntax

[d:] [Weg] BACKUP
[d:] [Weg] BACKUP ?
[d:] [Weg] BACKUP [Dateibez [+Dateibez ...]] [D:] [/X...]

dabei bedeutet:

d: vor dem Programmnamen bestimmt das aufzurufende Laufwerk

Weg vor dem Programmnamen bestimmt das aufzurufende Verzeichnis

Dateibez ein oder mehrere Namen von zu sichernden Dateien

D: Name des Ziellaufwerkes für die Sicherungsdatei

/X... Kombination aus folgenden Zusatzeingaben:

- **/A** bedeutet, daß die zu sichernden Dateien hinter bereits vorhandene Dateien auf der Sicherungsdiskette gespeichert werden.
- **/B[:Datum]** bedeutet, daß nur Dateien mit einem Herstellungsdatum vor dem eingegebenen Datum ge-sichert werden.
- **/D[:Datum]** bedeutet, daß nur Dateien mit einem Herstellungsdatum nach dem eingegebenen Datum ge-sichert werden.
- **/E** bedeutet, daß die bezeichneten Dateien nicht gesichert werden.
- **/F** bedeutet, daß die Zieldiskette automatisch ohne vorherige Anfrage formatiert wird.

/G
/S bedeutet, daß alle Dateien der Platte in allen Verzeichnissen gesichert werden (gilt für beide Zusätze).

/L bedeutet, daß das interne Verzeichnis der Sicherungsdatei angezeigt wird.

/M
/W bedeutet, daß nur Dateien kopiert werden, die seit dem letzten Sichern verändert wurden (gilt für beide Zusätze).

/N bedeutet, daß die Zieldiskette nicht formatiert wird.

/Q bedeutet, daß jede Datei vor dem Kopieren angezeigt und abgefragt wird.

/R bedeutet, daß mit jeder Meldung ein akustisches Signal erzeugt wird.

/T bedeutet, daß alle Dateien mit dem heutigen Datum kopiert werden.

/V bedeutet, daß die kopierten Dateien anschließend auf fehlerhafte Sektoren überprüft werden.

Bemerkungen

Das Programm BACKUP kann im *Netzwerk* nur **mit** Einschränkungen benutzt werden. Die Zieldiskette wird nicht formatiert.

Das Dienstprogramm BACKUP *sichert* schnell und einfach Ihre Dateien von einer *Diskette* oder *Winchester*. Mit dem Programm können Dateien zwischen allen Peripheriegeräten, mit Ausnahme von CON oder PRN, die ASCII- und Binärzeichen lesen können, kopiert werden. BACKUP sichert sowohl alle Dateien einer Platte (unabhängig von der Anzahl), als auch nur bestimmte.

Die zu sichernden Dateien werden auf dem Ziellaufwerk mit ihrem eigenen Namen, aber in einem anderen Format gespeichert. Das Programm BACKUP schreibt zusätzlich einen Dateikopf mit speziellen Informationen für das Programm RESTORE. Daher können Sie die Dateien nicht mehr wie gewohnt benutzen, sondern nur noch mit RESTORE. Außerdem wird eine separate Datei mit allgemeinen Informationen angelegt. Die Sicherungskopien können sich über mehrere Disketten erstrecken. Die Disketten werden dann intern durchnumeriert und erhalten damit einen eigenen Namen (Volume). Das Programm fordert Sie jeweils auf, die Disketten zu wechseln.

Programmaufruf

Sie können das Programm BACKUP mit verschiedenen Befehlen aufrufen. Mit Eingabe des Fragezeichens rufen Sie nur das Menü mit allen Befehlserklärungen auf. Geben Sie ein

BACKUP ?

und es erscheint folgendes:

BACKUP Version 3.04
Copyright(C) 1985 Zenith Data Systems Corporation

The BACKUP utility is designed to take any number of source files and transfer them to another disk for archive purposes. The backed up files may extend across several volumes, and should not be used without restoring them using the RESTORE program. The destination can be any block device, but unless the /A switch is used, the root directory will be destroyed.

Syntax: BACKUP [<filespec>[+<filespec>...] <d:> [</x>...]]

Switches: Default state: /F off, /Q off, and /V off.

/A Append to backup disk.
/B On or before date. /B:<mm-dd-yy>
/D On or after date. /D:<mm-dd-yy>
/E Exception files.
/F Format silent.
/G or /S Global subdirectories.
/M or /W Written files only.
/N No formatting.
/Q Query each.
/R Ring bell to prompt.
/T Today's date.
/V Verify files.

BACKUP

Für eine Befehlseingabe ist das Programm erneut aufzurufen. Dabei können Sie zwischen zwei Möglichkeiten wählen:

- Sie geben alle Zusätze in der Befehlszeile ein.
- Sie geben die Zusätze auf eine entsprechende Anfrage ein.

Rufen Sie das Programm ohne weitere Zusätze auf, erscheint der Programmname und das *Prompt* des Programmes ein „Größer-als-Zeichen“:

```
                BACKUP Version 3.04
Copyright(C) 1985 Zenith Data Systems Corporation
>
```

Sie können jetzt entsprechende Befehle eingeben. Nach jeder Ausführung erscheint wieder das Prompt, und das Programm wartet auf weitere Eingaben. Sie verlassen das Programm erst mit der Eingabe von **RETURN** oder der Tastenkombination **CTRL-C.** Diese Art der Eingabe hat den Vorteil, daß Sie den Programmnamen nicht immer mit eingeben müssen. Die Abarbeitung der Befehle erfolgt genauso, als wenn die Eingaben komplett mit dem Befehlsaufruf erfolgt wären.

Haben Sie das Programm aufgerufen und der Kopiervorgang hat begonnen, zeigt das Programm laufend den jeweiligen Arbeitsgang an. Zwischendurch kann es erforderlich sein, daß Sie Disketten austauschen müssen, oder daß Sie für die weitere Bearbeitung bestimmte Fragen des Programmes beantworten müssen.

Das Programm BACKUP fragt zuerst das Ziellaufwerk ab. Dabei wird unterschieden zwischen:

- Diskettenlaufwerk
- Winchesterlaufwerk
- Netzwerk- oder umgeleitetem Laufwerk

Disketten werden immer formatiert, haben Sie nicht den Zusatz /N eingegeben. Auf einem Winchesterlaufwerk wird ein neues Verzeichnis mit dem Namen \BACKUP angelegt. Ist dieses Verzeichnis schon vorhanden, werden dort gespeicherte Dateien gelöscht (Ausnahme: Zusatz /A). Laufwerke im Netzwerk oder Laufwerke, die mit den Befehlen ASSIGN oder SUBST umgeleitet wurden, können nicht formatiert werden. Hier werden die gesicherten Dateien immer hinter schon vorhandene kopiert.

BACKUP benötigt zum Formatieren der Disketten das Programm FORMAT. Dies muß entweder im aktuellen Verzeichnis vorhanden sein oder in einem Verzeichnis, daß mit dem Befehl PATH benannt wurde.

Benennen Sie ein Diskettenlaufwerk als Zielort, werden Sie gefragt:

Format backup disk (Y/N)?	Soll die Sicherungsdiskette formatiert werden (J/N)?

Diese Frage erscheint für jede *Sicherungsdiskette* neu. Geben Sie ein **N**(ein) ein, werden die Daten sofort auf die Diskette kopiert. Arbeiten Sie in einem Netzwerk, müssen Sie ein N(ein) eingeben. Geben Sie ein **Y** für Ja ein, erscheint folgende weitere Frage:

Format single or double sided (S/D)?	Soll die Diskette ein- oder zweiseitig formatiert werden (S/D)?

Es ist nur die Eingabe von **S** für einseitig oder **D** für doppelseitig erlaubt. Je nach Eingabe wird die Diskette nun formatiert. Danach werden die Dateien kopiert.

Sämtliche bestimmten Dateien werden auf die Sicherungsdiskette kopiert. Haben Sie mehr Dateien bestimmt, als die Diskette an Speicherkapazität besitzt, müssen Sie die Diskette austauschen. Es erscheint folgende Aufforderung:

Insert another disk in drive D: for backup and press RETURN when ready or any key to abort	Legen Sie eine neue Diskette in Laufwerk D: und betätigen Sie RETURN oder eine Taste zum Abbruch

Legen Sie eine leere Diskette in das Laufwerk, und betätigen Sie die Taste **RETURN,** damit weiter kopiert wird oder irgendeine andere Taste, um den Vorgang abzubrechen. Haben Sie eine beliebige Taste betätigt, erscheint die Frage:

Are you sure you want to abort (Y/N)?	Wollen Sie das Programm abbrechen oder fortsetzen? (A/F)

Beenden Sie das Programm nicht, erscheint die Aufforderung für die neue Diskette noch einmal. Auch für diese neue Diskette müssen Sie die Frage nach dem Formatierungsformat beantworten.

Während des Kopiervorganges wird jede einzelne Quelldatei auf dem Bildschirm angezeigt. Dies geschieht in der Form:

```
C:Dateiname.Zus
C:Dateiname.Zus
 .
 .
C:Dateiname.Zus
```

Zum Schluß markiert BACKUP jede gesicherte Datei. Diesen Merker benötigen Sie für die Zusätze /M oder /W. Bei einem erneuten Aufruf des Programmes BACKUP erkennt dieses daran alle Dateien, die seit der letzten Sicherung verändert wurden. Achten Sie daher darauf, daß die Ursprungsdiskette nicht mit einem Schreibschutz versehen ist.

Quelldatei (zu sichernde Datei)

Nach dem Programmnamen BACKUP muß von Ihnen mindestens eine *Quelldatei* bezeichnet werden. Der Name der Quelldatei darf dabei die Zeichen zur Zusammenfassung mehrerer Dateien („*“ und „ ? “) enthalten.

Es ist auch möglich, mehrere Quelldateien einzugeben. Jeder Name darf dann die Zeichen „*“ und „ ? “ enthalten. Die Quelldateien müssen durch ein Pluszeichen (+) miteinander verbunden werden. Andere Zeichen dürfen nicht eingegeben werden, ebenso dürfen keine Leerräume zwischen den Dateinamen vorhanden sein. Jedem einzelnen Dateinamen darf eine Laufwerkbezeichnung und/oder ein Verzeichnisname vorangestellt werden.

Die Informationen im Kopf der gesicherten Datei enthalten auch den vollständigen Verzeichnisnamen des Ursprungslaufwerkes. Lediglich der Laufwerkname wird nicht mit gespeichert.

Ziellaufwerk

Nach der Quelldatei erfolgt die Eingabe eines *Ziellaufwerkes.* Dies darf jeder gültige Laufwerknamen sein, jedoch nie ein Verzeichnisname. Der Name des Ziellaufwerkes wird in der Befehlszeile vom Namen der Quelldatei durch ein Leerfeld getrennt. Das Programm gibt einer Zieldiskette intern eine Nummer. Die Numerierung beginnt bei 01 für die erste Diskette. Diese Nummer wird vom Programm RESTORE benötigt. Haben Sie mehr Dateien bestimmt, als auf einer Diskette gespeichert werden können, werden Sie aufgefordert, eine neue Diskette in das Laufwerk zu legen. Die neue Diskette wird ebenfalls automatisch numeriert.

Zusatzeingaben

Über verschiedene *Zusatzeingaben* haben Sie die Möglichkeit, den Programmablauf zu beeinflussen. Alle Eingaben werden zum Schluß der Befehlszeile eingegeben. Sie werden von vorhergehenden Eingaben mit einem Schrägstrich (/) getrennt. Geben Sie keine Zusätze ein, führt das Programm folgende Funktionen automatisch aus:

- Anfrage zur Formatierung der Zieldiskette – wird mit dem Zusatz /F oder /N geändert.
- keine Anfrage vor dem Kopieren jeder Datei – wird mit dem Zusatz /Q geändert.
- keine Fehlerprüfung nach dem Kopieren – wird mit dem Zusatz /V geändert.

/A - Dateien anfügen

Mit diesem Zusatz weisen Sie das Programm an, die Zieldiskette nicht zu formatieren. Kopieren Sie die Dateien auf ein Winchester-laufwerk, wird das Verzeichnis \BACKUP nicht vorher gelöscht. Werden Sie aufgefordert, eine Zieldiskette in das Laufwerk zu legen, kopiert BACKUP die Dateien sofort. Achten Sie daher immer darauf, daß die Zieldisketten über genügend freie Speicherkapazität besitzen. Rufen Sie das Programm z. B. mit folgendem Befehl auf:

```
BACKUP *.TXT A:/A
```

Alle Dateien mit dem Zusatz TXT werden jetzt auf das Laufwerk A: kopiert und hinter bereits vorhandene Dateien gespeichert.

/B - vor Datum

Mit diesem Zusatz werden die zu sichernden Dateien weiter selektiert. Das Programm vergleicht das Datum der Datei im Inhaltsverzeichnis mit dem Datum, daß Sie nach diesem Zusatz eingeben. Die Dateien mit einem *früheren* Datum werden kopiert, andere Dateien werden übergangen. Geben Sie nach dem Zusatz kein Datum ein, ist das momentane Systemdatum das Vergleichsdatum.

Sollen nur die Dateien gesichert werden, deren Herstellungs- oder Änderungsdatum *früher* ist, als das momentane (heutige) Datum, geben Sie ein:

```
/B
```

Das Programm vergleicht jetzt das Systemdatum mit dem Datum der ausgewählten Dateien und kopiert nur noch die Dateien, deren Datum *kleiner* ist.

Geben Sie den Zusatz mit einem Datum ein, erfolgt die Datumseingabe in der Schreibweise, die Sie mit dem Befehl COUNTRY bestimmt haben. Datum und Zusatz werden durch einen Doppelpunkt getrennt. Haben Sie das Betriebssystem mit dem Befehl COUNTRY=49 gebootet, erfolgt die Eingabe in der Form

/B:tt-mm-jj

dabei bedeutet:

tt	Eingabe des Tages mit einer Zahl von 1 bis 31
mm	Eingabe des Monats mit einer Zahl von 1 bis 12
jj	Eingabe des Jahres mit einer Zahl von 00 bis 99

Die einzelnen Zahlen des Datums müssen durch einen Bindestrich getrennt werden. Andere Zeichen sind nicht erlaubt.

Das Programm kopiert dann nur noch die ausgewählten Dateien, deren Datum *kleiner* ist als das eingegebene.

/D - nach Datum

Mit diesem Zusatz werden die zu sichernden Dateien weiter selektiert. Das Programm vergleicht das Datum der Datei im Inhaltsverzeichnis mit dem Datum, daß Sie nach diesem Zusatz eingeben. Die Dateien mit einem späteren Datum werden kopiert, andere Dateien werden übergangen. Geben Sie nach dem Zusatz kein Datum ein, ist das momentane Systemdatum das Vergleichsdatum.

Sollen nur die Dateien gesichert werden, deren Datum der Herstellung oder letzten Änderung später ist als das momentane (heutige) Datum, geben Sie ein:

/D

Das Programm vergleicht jetzt das *Systemdatum* mit dem Datum der ausgewählten Dateien und kopiert nur noch die Dateien, deren Datum größer ist.

Geben Sie den Zusatz mit einem Datum ein, erfolgt die Datumseingabe in der Schreibweise, die Sie mit dem Befehl COUNTRY bestimmt haben. Datum und Zusatz werden durch einen Doppelpunkt getrennt. Haben Sie das Betriebssystem mit dem Befehl COUNTRY=49 gebootet, erfolgt die Eingabe in der Form

/D:tt-mm-jj

dabei bedeutet:

tt	Eingabe des Tages mit einer Zahl von 1 bis 31
mm	Eingabe des Monats mit einer Zahl von 1 bis 12
jj	Eingabe des Jahres mit einer Zahl von 00 bis 99

Die einzelnen Zahlen des Datums müssen durch einen Bindestrich getrennt werden. Andere Zeichen sind nicht erlaubt.

Das Programm kopiert dann nur noch die ausgewählten Dateien, deren Datum *größer* ist als das eingegebene.

/E - Datei nicht sichern

Mit diesem Zusatz werden *Dateien* bezeichnet, die nicht kopiert werden sollen. Dies ist hilfreich, wenn Sie mehrere Dateien mit dem Stern oder Fragezeichen zusammenfassen. Alle Dateien, die nach dem Zusatz eingegeben werden, sichert das Programm nicht. In dieser Eingabe dürfen auch die Zeichen zur Dateizusammenfassung verwendet werden. Geben Sie mehrere Dateinamen ein, müssen Sie die Namen durch ein Pluszeichen (+) verbinden. Andere Zeichen dürfen zwischen den Namen nicht eingegeben werden, auch keine Leerzeichen. Die Eingabe erfolgt in der Form

/E:Dateiname[+Dateiname...]

Der Zusatz und die Dateinamen werden durch einen Doppelpunkt (:) getrennt. Von den bezeichneten Quelldateien werden jetzt nur noch die Dateien kopiert, die durch den Zusatz /E nicht erfaßt wurden.

Ein Beispiel:

BACKUP C:*.* a:TXTBACK /E:*.DAT

Es werden jetzt alle Dateien des Laufwerkes C: kopiert, mit Ausnahme der Dateien deren Zusatz „DAT" ist.

/F - formatiere automatisch

Das Programm kann die zu *sichernden Dateien* nur auf korrekt formatierte Disketten kopieren. Es *formatiert* daher jede Diskette neu. Haben Sie auf eine entsprechende Aufforderung hin eine neue Diskette in das Laufwerk gelegt, wird diese Diskette erst formatiert.

Der Zusatz /F ändert den Formatierungsablauf. Da jetzt automatisch formatiert werden soll, entfällt die entsprechende Anfrage. Es erscheint sofort die Frage nach dem Format:

Format single or double sided (S/D)?	Soll die Diskette ein- oder zweiseitig formatiert werden (J/N)?

Es ist nur die Eingabe von **S** für einseitig oder **D** für doppelseitig erlaubt. Je nach Eingabe wird die Diskette nun formatiert. Danach werden die Dateien kopiert.

/G oder /S - kopiere alle Verzeichnisse

Normalerweise kopiert das Programm nur die bezeichneten Dateien des momentanen Verzeichnisses. Alle anderen Dateien, selbst wenn sie mit dem eingegebenen Quelldateinamen übereinstimmen, bleiben unberücksichtigt. Durch Eingabe des Zusatzes /G oder /S wird das Programm angewiesen, *alle Verzeichnisse* ab dem momentanen Verzeichnis abzusuchen. Geben Sie den Befehl vom Rootdir aus ein, wird die gesamte Platte abgesucht. Geben Sie den Befehl von einem Subdir aus ein, werden nur die Verzeichnisse ab dem momentanen Verzeichnis abgesucht. Alle höheren Verzeichnisse bleiben weiterhin unberücksichtigt.

Geben Sie als Dateinamen ein „ *.* ", werden alle Dateien des Verzeichnisses kopiert. Rufen Sie das Programm vom Rootdir aus mit diesem Dateinamen und dem Zusatz /G oder /S auf, wird die gesamte Platte gesichert.

/M oder /W - nur veränderte Dateien

Mit diesem Zusatz werden die zu sichernden Dateien weiter selektiert. Es werden nur noch die Dateien gesichert, die seit der *letzten Sicherung* verändert wurden.

Im Inhaltsverzeichnis jeder Datei ist durch das Betriebssystem ein *Bit* reserviert zur Kennzeichnung der Datei (s. a. Befehl FORMAT). Dieses Bit wird vom Programm BACKUP gelöscht. Verändern Sie die Datei anschließend, wird dieses Bit vom Betriebssystem wieder gesetzt. Geben Sie den Zusatz /M oder /W ein, wird dieses Bit vom Programm abgefragt. Es werden nur noch Dateien mit gesetztem Bit kopiert.

/N - keine Formatierung

Mit diesem Zusatz weisen Sie das Programm an, die *Zieldisketten* nicht zu formatieren. Werden Sie aufgefordert, eine Zieldiskette in das Laufwerk zu legen, werden anschließend die Dateien sofort kopiert. Die Fragen zur Formatierung werden unterdrückt. Achten Sie daher darauf, daß die Zieldisketten immer richtig formatiert sind und genügend freie Speicherkapazität für die Sicherungsdatei haben.

/Q - Datei abfragen

Nach Eingabe dieses Zusatzes zeigt das Programm erst jede bezeichnete *Quelldatei* einzeln an. Sie werden zu jeder Datei gefragt:

Backup ... (Y/N)?	Soll diese Datei kopiert werden?

Haben Sie die *Quelldateien* global bezeichnet, ist dieser Zusatz sehr hilfreich. Er zeigt jede Datei einzeln an. Sie können dann entscheiden, ob die Datei kopiert werden soll. Geben Sie ein **Y** für Ja ein, wird die Datei anschließend kopiert. Geben Sie ein **N(ein)** ein, wird die Datei aus der Liste der zu kopierenden Dateien wieder gestrichen.

/R - Summer ertönen lassen

Der Zusatz /R gibt zu jeder Meldung ein akustisches Signal. Dies erfolgt z. B. immer mit der Meldung, die Diskette zu tauschen. Ebenfalls kann damit Ihre Aufmerksamkeit bei einer Fehlermeldung geweckt werden.

/T - heutiges Datum

Mit diesem Zusatz werden die zu sichernden Dateien weiter selektiert. Das Programm vergleicht das Datum der Datei im Inhaltsverzeichnis mit dem heutigen Datum. Es werden nur noch die bezeichneten Dateien kopiert, deren Datum dem Systemdatum *entspricht,* andere Dateien werden übergangen.

/V - Dateien überprüfen

Mit diesem Zusatz weisen Sie das Programm an, jede Datei sofort nach dem Kopiervorgang mit der *Quelldatei* zu vergleichen. Das Programm zeigt Ihnen dies durch die Meldung

Verifying ...	Überprüfe Datei

an. Ist die kopierte Datei feherfrei, wird die nächste Datei gesichert. Findet das Programm einen Fehler, erscheint die Frage:

Verify error, try BACKUP again (Y/N)?	Fehler gefunden, soll erneut kopiert werden?

Durch die Eingabe von **Y** für Ja, wiederholen Sie den Kopier- und anschließenden Prüfvorgang. Geben Sie ein **N**(ein) ein, wird mit der nächsten Datei fortgefahren. Die fehlerhafte Datei wird nicht korrigiert.

Beispiele

Die folgenden Beispiele zeigen Ihnen verschiedene Befehlseingaben für das Programm BACKUP.

Beispiel 1

Sie wollen mehrere einzelne Dateien in einer Sicherungsdatei zusammenfassen. Die Quelldateien sind nicht alle auf demselben Laufwerk. Sie geben ein:

```
BACKUP A:*.BAS+B:KAP.*+C:???X.COM F: /R /N
```

Vom Laufwerk A: werden alle Dateien mit dem Zusatz .BAS ausgewählt. Vom Laufwerk B: werden alle Dateien mit dem Namen KAP ausgesucht, unabhängig vom Namenszusatz. Vom Laufwerk C: werden alle Dateien, deren viertes Zeichen des Namens ein X ist und die den Zusatz .COM haben, ausgewählt. Die Sicherungsdatei wird auf das Laufwerk F: geschrieben.

Es wurden die Zusätze /R und /N eingegeben. Der Zusatz /R gibt alle Meldungen mit einem Summton aus. Der Zusatz /N bedeutet, daß das Laufwerk F: nicht neu formatiert wird.

Beispiel 2

Sie haben auf dem Laufwerk F: Dateien mit verschiedenen Zusätzen gespeichert und wollen gezielt bestimmte Dateien sichern. Auf dem Laufwerk sind Dateien mit den Zusätzen .LTR, .TXT, und .DAT vorhanden. Die Dateien mit dem Zusatz .DAT sollen nicht gesichert werden. Geben Sie ein:

```
BACKUP F:*.* A: /G /E:*.DAT
```

Mit der Eingabe von „*.*“ bestimmen Sie alle Dateien unabhängig vom Namen oder Zusatz. Da der Zusatz /G mit eingegeben wurde, werden alle Dateien des Laufwerkes erfaßt. Die Sicherungsdatei wird auf das Laufwerk A: geschrieben. Der Zusatz /E:*.DAT bestimmt, daß alle Dateien mit dem Zusatz .DAT nicht kopiert werden.

Beispiel 3

Die Eingabe des letzten Beispiels läßt sich durch den Zusatz /Q leicht variieren. Jetzt werden sämtliche bezeichneten Dateien vorher abgefragt, und Sie haben die Möglichkeit, einzelne Dateien zu löschen. Geben Sie ein:

```
BACKUP F:*.* A:TEXTBACK /G /E:*.DAT
```

Es erscheint jetzt nach einem kurzen Moment für jede Datei die Frage:

BACKUP F:... (Y/N)?	Soll Datei ... gesichert werden?

Geben Sie ein **Y** für Ja ein, wird die Datei anschließend gesichert. Geben Sie ein **N**(ein) ein, wird die Datei aus der Liste gestrichen.

Beispiel 4

Sie können das Programm BACKUP auch ohne weitere Zusätze in der Befehlszeile aufrufen. Sie geben dann die entsprechenden Befehle nacheinander ein. Rufen Sie das Programm auf:

```
BACKUP
```

Es erscheint der Programmname und das Prompt (>):

```
                  BACKUP Version 3.04
Copyright(C) 1985 Zenith Data Systems Corporation
>
```

Sie können jetzt mehrere Befehle eingeben, ohne immer wieder das Programm neu aufrufen zu müssen.

Sie können z. B. erst eine neue Sicherungsdatei erstellen. Alle Dateien mit dem Zusatz .TXT sollen von verschiedenen Laufwerken zusammen gesichert werden. Geben Sie ein:

```
C:*.TXT+D:*.TXT+E:*.TXT+F:*.TXT A: /V
```

Sind alle Dateien gesichert worden, erscheint wieder das Prompt. Sie können jetzt weitere Befehle eingeben oder das Programm mit Eingabe von **RETURN** nach dem BACKUP-Prompt (>) verlassen. Es erscheint dann wieder das DOS-Prompt (C>).

Sichern mit einer Batchdatei

Sie sollten täglich Ihre Dateien sichern, mit denen Sie arbeiten. Dies ist die beste (und einzige) Versicherung gegen Datenverlust. Das Programm BACKUP hilft Ihnen dabei. Benutzen Sie es in einer Bachtdatei, können Sie weitere Vorteile des Programmes nutzen.

Das Programm übergibt bei seinem Ende dem Betriebssystem immer einen Wert. Mit dem Befehl ERRORLEVEL können Sie diesen Wert in einer Batchdatei abfragen und den weiteren Programmablauf steuern. Die folgende Tabelle zeigt eine Übersicht.

Tabelle 11.1 Ausgangswerte des Programmes BACKUP

WERT	BEDEUTUNG
0	Programm fehlerfrei beendet
1	keine Dateien gefunden
2	Fehler durch Netzwerk
3	vom Bediener abgebrochen
4	durch einen Fehler abgebrochen

Benutzen Sie BACKUP täglich, sollten Sie für jeden Arbeitstag eine neue Sicherungsdiskette nehmen und mit dem jeweiligen Wochentag beschriften. Dafür könnten Sie die folgende Batchdatei mit dem Namen SICHERN.BAT benutzen:

```
ECHO OFF
CLS
IF "%1" == "" GOTO FEHLER1
IF "%2" == "" GOTO FEHLER2
ECHO Haben Sie eine Diskette in das Laufwerk %2 gelegt?
PAUSE
BACKUP %1*.* %2 /G /T /V /R
IF ERRORLEVEL 4 ECHO Programm durch einen Fehler abgebrochen
IF ERRORLEVEL 3 ECHO Programm vom Bediener abgebrochen
IF ERRORLEVEL 2 ECHO Programm durch Netzwerkfehler abgebrochen
IF ERRORLEVEL 1 ECHO Keine Dateien zum Sichern gefunden
IF ERRORLEVEL 0 ECHO Alle Dateien fehlerfrei gesichert
GOTO ENDE
:FEHLER1
ECHO Es wurde kein Ursprungslaufwerk eingegeben
GOTO ENDE
:FEHLER2
ECHO Es wurde kein Ziellaufwerk eingegeben
:ENDE
```

Rufen Sie die Datei mit folgender Befehlszeile auf:

```
[d:] [Weg] SICHERN d1: d2:
```

dabei bedeutet:

d:	vor dem Programmnamen bestimmt das aufzurufende Laufwerk
Weg	vor dem Programmnamen bestimmt das aufzurufende Verzeichnis
d1:	Name des Ursprungslaufwerkes mit den zu sichernden Dateien
d2:	Name des Ziellaufwerkes für die gesicherten Dateien

BACKUP

Bemerkungen

Die Bachtdatei sichert alle Dateien (Zusatz /G) des Ursprungslaufwerkes mit dem heutigen Datum (Zusatz /T). Die Kopien werden noch einmal mit dem Original verglichen (Zusatz /V). Bei jeder Textausgabe ertönt der Summer (Zusatz /R). Die Datei muß mit zwei Zusätzen aufgerufen werden: dem Ursprungslaufwerk und dem Ziellaufwerk. Die Eingabezeile wird zuerst daraufhin überprüft. Dann erfolgt ein Hinweis für das Ziellaufwerk. Damit Sie die Datei jederzeit problemlos aufrufen können, wird in das Rootdir gewechselt. Jetzt werden alle Dateien gesichert. Anschließend wird der Ausgangswert des Programmes abgefragt und angezeigt.

Fehlermeldungen

All files in directory will be destroyed — Alle Dateien des Verzeichnisses werden gelöscht

ERKLÄRUNG:
Sie haben als Ziel ein Winchesterlaufwerk bestimmt. Das Verzeichnis \BACKUP enthält bereits Dateien. Fahren Sie jetzt mit dem Programm fort, werden diese gelöscht. Sie können das Programm mit der Tastenkombination **CTRL-C** abbrechen und mit dem Zusatz /A erneut aufrufen.

All data in the root directory will be destroyed — Alle Dateien der Diskette werden gelöscht

ERKLÄRUNG:
Sie haben als Ziel ein Diskettenlaufwerk bestimmt. Die Diskette enthält bereits Dateien. Fahren Sie jetzt mit dem Programm fort, werden diese gelöscht. Sie können das Programm mit der Tastenkombination **CTRL-C** abbrechen und mit dem Zusatz /A erneut aufrufen.

Cannot execute format	Das Programm FORMAT kann nicht aufgerufen werden

ERKLÄRUNG:
Die Zieldiskette soll formatiert werden. BACKUP kann jedoch die Datei FORMAT.EXE nicht finden. Kopieren Sie die Datei in das aktuelle Verzeichnis, oder benennen Sie das Verzeichnis des Programmes mit dem Befehl PATH.

Conflicting switches /F and /N specified	Die Zusätze /F und /N wurden zusammen eingegeben.

ERKLÄRUNG:
Sie haben den Zusatz /F, zum Formatieren der Diskette, mit dem Zusatz /N, zum Nichtformatieren der Diskette, zusammen eingegeben. Geben Sie den Befehl mit nur einem Zusatz neu ein.

Format failure, insert another disk and press RETURN, or press any other key to abort	Fehler bei der Formatierung. Legen Sie eine andere Diskette ins Laufwerk und betätigen Sie RETURN, oder eine andere Taste zum Abbruch

ERKLÄRUNG:
Die neue Diskette kann nicht formatiert werden. Möglicherweise ist sie beschädigt. Nehmen Sie eine andere Diskette, und überprüfen Sie später die beschädigte Diskette.

Insert another disk in drive D: for backup, and press RETURN when ready, or any other key to abort	Legen Sie eine neue Diskette in das Laufwerk D: und betätigen Sie RETURN, oder eine andere Taste zum Abbruch

ERKLÄRUNG:
Die momentane Diskette ist voll. Legen Sie eine leere Diskette in das Laufwerk, damit das Programm fortfahren kann.

BACKUP

Invalid date in switch	Falsche Datumseingabe

ERKLÄRUNG:
Das Datum, das Sie mit dem Zusatz /A oder /B eingegeben haben, ist falsch. Überprüfen Sie Ihre Eingabe.

Invalid drive specified	Falsche Laufwerkbezeichnung

ERKLÄRUNG:
Sie haben eine falsche Laufwerkbezeichnung eingegeben. Das Laufwerk ist nicht vorhanden oder wird vom Programm nicht erkannt. Es können die Laufwerke A: bis H: benannt werden.

Invalid exception file specifications	Falsche Dateibezeichnung für die mit /E bezeichneten Dateien

ERKLÄRUNG:
Sie haben den Zusatz /E ohne Dateiname oder den Dateinamen falsch eingegeben. Überprüfen Sie Ihre Eingabe.

Invalid file name	Falscher Dateiname

ERKLÄRUNG:
Der von Ihnen benannte Dateiname enthält ungültige Zeichen. Überprüfen Sie Ihre Eingabe.

Invalid selection file specifications	Falsche Eingabe der Dateiauswahl

ERKLÄRUNG:
Sie haben Dateinamen oder Zusätze falsch eingegeben. Überprüfen Sie Ihre Eingabe.

Invalid switch /X specified	Falsche Zusatzeingabe /X benannt

ERKLÄRUNG:
Sie haben nach dem Schrägstrich (/), der den folgenden Buchstaben als Zusatzeingabe definiert, ein falsches Zeichen eingegeben. Überprüfen Sie Ihre Eingabe.

No files selected	Keine Dateien benannt

ERKLÄRUNG:
Sie haben in der Befehlszeile keine Dateinamen benannt, oder die eingegebenen Namen erfüllen nicht die Bedingungen der Zusatzeingaben. Diese Meldung erscheint auch, wenn Sie auf eine Frage des Zusatzes /Q mit **N**(ein) antworten.

Not able to back up this time	Die Datei kann momentan nicht gesichert werden

ERKLÄRUNG:
Sie versuchen eine Datei zu sichern, die im Netzwerk von einem anderen Benutzer für alle Zugriffe gesperrt wurde. Warten Sie, bis die Sperre wieder aufgehoben ist.

Not enough parameters specified	Nicht alle Eingaben vorgenommen

ERKLÄRUNG:
Sie haben die Befehlszeile unvollständig abgeschlossen. Überprüfen Sie Ihre Eingaben.

Too many parameters specified	Zu viele Zusatzeingaben

ERKLÄRUNG:
Sie haben mehr Zusätze eingegeben, als vom Programm verarbeitet werden können. Überprüfen Sie Ihre Eingabe.

Too many subdirectories	Zu viele Verzeichnisse

ERKLÄRUNG:
Sie haben den Zusatz /G oder /S eingegeben. BACKUP muß zu viele Unterverzeichnisse absuchen. Dadurch wird der komplette Name für die einzelne Datei zu lang. Rufen Sie das Programm neu auf, dann von einem tieferen Verzeichnis aus. Dadurch wird der komplette Dateiname kürzer.

BACKUP

Verify error, try BACKUP again (Y/N)?	Prüffehler, soll die Datei erneut kopiert werden (J/N)?

ERKLÄRUNG:
Die zu sichernde Datei wurde nicht korrekt auf die Diskette geschrieben. Geben Sie ein **Y** für Ja ein, wird die Datei erneut kopiert. Geben Sie ein **N**(ein) ein, wird mit der nächsten Datei fortgefahren.

BOOTF Extern

Zweck

schaltet das (die) Winchesterlaufwerk(e) ab

Syntax

[d:] [Weg] BOOTF

dabei bedeutet:

d:	vor dem Programmnamen bestimmt das aufzurufende Laufwerk
Weg	vor dem Programmnamen bestimmt das aufzurufende Verzeichnis

Bemerkungen

Fast alle für den IBM-PC und andere kompatible Rechner geschriebene Programme arbeiten auch auf Ihrem Zenith-PC einwandfrei. Trotzdem kann es vereinzelt vorkommen, daß ein Programm nicht sofort fehlerfrei arbeitet. Dieses Programm wurde dann für einen PC mit Doppeldiskettenlaufwerk konzipiert oder verfügt über einen Kopierschutz auf der Diskette und erlaubt keine Kopie auf das Winchesterlaufwerk.

Ihr Zenith-PC verfügt über zahlreiche Routinen im Monitorprogramm, die die meisten Inkompatibilitäten zwischen einem Disketten- und Winchesterlaufwerk ausgleichen. Trotzdem kann es vorkommen, daß dies nicht immer gelingt. Auch besteht die Möglichkeit, daß sich das Diskettenlaufwerk nicht mehr ausschaltet und Ihr Gerät keine Befehle mehr ausführt. Sie müssen dann das Betriebssystem neu starten (booten). Für diese Programme besteht die Möglichkeit, das Winchesterlaufwerk abzuschalten.

BOOTF

Das Programm BOOTF, das auf der zweiten Systemdiskette enthalten ist, schaltet alle angeschlossenen Winchesterlaufwerke ab. Ein Zugriff auf die dort gespeicherten Daten ist dann nicht mehr möglich. Rufen Sie das Programm auf durch Eingabe von:

```
BOOTF
```

Das Programm schaltet das Winchesterlaufwerk ab und startet das System vom Laufwerk A: erneut. Achten Sie daher darauf, daß Sie in diesem Falle im Laufwerk A: eine bootfähige Diskette eingelegt haben.

Sie können dann nur noch das (die) Diskettenlaufwerke Ihres Gerätes benutzen. Das Winchesterlaufwerk bleibt bis zum nächsten Systemstart ausgeschaltet.

Das Programm BOOTF ist nur hilfreich bei problematischen Programmen auf einem Gerät mit Winchesterlaufwerk.

BREAK Intern

Zweck

aktiviert Programmunterbrechung durch CTRL-BREAK

Syntax

```
BREAK [ON]
BREAK [OFF]
```

dabei bedeutet:

ON	jede Eingabe wird auf die Tastenkombination überprüft.
OFF	nur bei Ein-/Ausgaben von Peripheriegeräten erfolgt eine Überprüfung auf die Kombination CTRL-BREAK

Bemerkungen

Mit diesem Befehl weisen Sie das Betriebssystem an, die Tastenkombination CTRL-C oder CTRL-BREAK zu beachten. Dies geschieht bei jedem Funktionsaufruf von MS-DOS, den ein Anwenderprogramm durchführt. Für Programme, die nicht mit MS-DOS arbeiten, kann dieser Befehl nicht benutzt werden.

Nach dem Systemstart ist die *Funktion* BREAK ausgeschaltet. Wollen Sie jetzt das laufende Programm abbrechen, geschieht dies erst bei einer Ein- oder Ausgabe eines Peripheriegerätes. Durch Eingabe von

```
BREAK ON
```

wird die *Programmunterbrechung* aktiviert, und der Abbruch durch CTRL-BREAK erfolgt sofort.

BREAK

Benutzt ein Anwenderprogramm die Tastenkombination **CTRL-BREAK** als gültige Eingabe, muß der Befehl

BREAK OFF

vorher eingegeben werden. Nur so kann die Eingabe vom Anwenderprogramm verarbeitet werden. Im anderen Falle würden Sie das Programm sofort verlassen und wären wieder im Betriebssystem.

Wissen Sie den momentanen Status nicht, geben Sie den Befehl ohne Zusatz ein:

BREAK

Es wird dann angezeigt:

BREAK is on	BREAK ist eingeschaltet

oder

BREAK is off	BREAK ist ausgeschaltet

Dieser Befehl kann auch in der Datei CONFIG.SYS abgearbeitet werden. Nähere Informationen dazu finden Sie im Kapitel 9.

CHDIR oder CD — Intern

Zweck

ändert das Verzeichnis, auf das zugegriffen werden soll, zeigt den Namen des Verzeichnisses

Syntax

CHDIR [D:] [Weg]
CD [D:] [Weg]

dabei bedeutet:

D: Name des Laufwerkes, auf dem das Verzeichnis geändert werden soll

Weg mehrere Verzeichnisse in Folge in der Form:

[\] [Verzeichnis [\Verzeichnis ...]]

Bemerkungen

Mit diesem Befehl können Sie das *momentane Verzeichnis* ändern. Sie können sich mit diesem Befehl durch das strukturierte Inhaltsverzeichnis der Platte bewegen. Sie können jederzeit zu jedem beliebigen Verzeichnis gehen. Ebenso ändern Sie mit diesem Befehl das Verzeichnis, in dem vom Betriebssystem auf eine Datei zugegriffen werden soll. Das kann das momentane Verzeichnis des Arbeitslaufwerkes sein, oder aber auch das Verzeichnis eines anderen Laufwerkes. Dafür muß dieses Laufwerk nach dem Befehlsnamen bezeichnet werden. Geben Sie den Befehl ohne weitere Zusätze ein, wird der Name des momentanen Verzeichnisses angezeigt. Ein Beispiel:

CHDIR

zeigt Ihnen den Namen des momentanen Verzeichnisses mit dem Laufwerknamen an. Sie können den Befehl auch in der Kurzform eingeben:

CD

CHDIR oder CD

Wollen Sie sich im Verzeichnis der Platte vom Rootdir wegbewegen, geben Sie den Befehl CD immer mit dem Namen des folgenden Verzeichnisses ein. Ein Beispiel:

```
CD SPEICHER
```

führt Sie vom momentanen Verzeichnis zum folgenden Verzeichnis mit dem Namen SPEICHER. Sie können auch mehrere Verzeichnisnamen hintereinander eingeben, jeweils durch einen umgekehrten Schrägstrich (\) getrennt, dann überspringen Sie einige Verzeichnisse. Ein Beispiel:

```
CD BASIC\SPEICHER\DATEI
```

führt Sie vom momentanen Verzeichnis über die Verzeichnisse BASIC und SPEICHER ins Verzeichnis DATEI.

Geben Sie nach dem Befehl sofort den Namen eines Verzeichnisses ein, so gehen Sie schrittweise vom momentanen Verzeichnis weiter. Geben Sie nach dem Befehl erst den umgekehrten Schrägstrich (\) ein, springen Sie zum Rootdir und gehen dann von dort aus schrittweise weiter. Dies geschieht unabhängig vom momentanen Verzeichnis aus. Dies gilt ebenso, wenn Sie den Befehl nur mit Schrägstrich eingeben. Ein Beispiel:

```
CD \
```

Sie springen direkt zum Rootdir. Da es als einziges Verzeichnis keinen eigenen Namen hat, wird es immer mit einem einzelnen umgekehrten Schrägstrich bezeichnet.

Die beiden ersten Einträge eines *Subdir* werden von MS-DOS bei der Neuanlage des Verzeichnisses sofort geschrieben. Sie enthalten den Namen des Verzeichnisses (Eintrag mit einem Punkt) und den Namen des Verzeichnisses, von dem aus das Verzeichnis angelegt wurde (Eintrag mit zwei Punkten). Diese Einträge können mit dem Befehl CD zusammen benutzt werden. Geben Sie den Befehl

```
CD .
```

ein, wird der Name des momentanen Verzeichnisses angezeigt (dasselbe wie CD).

Geben Sie den Befehl ein:

CD ..

Sie Wechseln zum vorherigen Verzeichnis zurück. Nur mit dieser Eingabe können Sie sich im System des Plattenverzeichnisses schrittweise bis zum Rootdir zurückbewegen. Dies ist erforderlich, wollen Sie z. B. auf einen anderen Ast des Verzeichnisses überwechseln. Ein Beispiel:

CD ..\..\TEXT\DATEI

Sie gehen vom momentanen Verzeichnis zweimal je ein Verzeichnis zurück und dann über das Verzeichnis TEXT zum Verzeichnis DATEI.

Alle Beispiele bezogen sich bislang auf das Arbeitslaufwerk. Diese Befehle gelten sinngemäß auch für jedes andere Laufwerk. Geben Sie dann nach dem Befehlsnamen die Laufwerkbezeichnung ein und anschließend den oder die Namen von Verzeichnissen. Ein Beispiel:

CD F:\BUCH\KAPITEL

Jetzt wird auf dem Laufwerk F: das Verzeichnis KAPITEL, das vom Verzeichnis BUCH angelegt wurde, zum Arbeitsverzeichnis des Laufwerkes F:. Dies ist immer dann von Bedeutung, wenn Sie vom momentanen Laufwerk aus auf ein anderes Laufwerk zugreifen wollen. MS-DOS greift auch bei anderen Laufwerken, als dem momentanen, immer auf das dortige *Arbeitsverzeichnis* zu. Dies ist entweder das zuletzt benutzte Verzeichnis, oder das mit dem Befehl CD benannte Verzeichnis.

Geben Sie den Befehl CD mit einem falschen Verzeichnisnamen ein oder mit einem Schreibfehler, erscheint die Fehlermeldung:

Invalid directory — Falsches Verzeichnis

Überprüfen Sie dann Ihre Eingabe, und wiederholen Sie den Befehl korrekt.

CHKDSK Extern

Zweck

überprüft eine Platte oder Datei und korrigiert Fehler

Syntax

[d:] [Weg] CHKDSK [D:] [Dateiname] [/X]

dabei bedeutet:

d:	vor dem Programmnamen bestimmt das aufzurufende Laufwerk
Weg	vor dem Programmnamen bestimmt das aufzurufende Verzeichnis
D:	Name des Laufwerkes mit der zu prüfenden Platte
Dateiname	Name der zu überprüfenden Datei
/X	Kombination aus folgenden Zusatzeingaben:
	/F korrigiert gefundene Fehler
	/V zeigt laufend die überprüfte Datei und gefundene Fehler an

Bemerkungen

Das Programm CHKDSK kann im *Netzwerk* **nicht** benutzt werden.

Das Programm CHKDSK (**Check Disk** = überprüfe Platte) *liest* das Inhaltsverzeichnis der Platte und überprüft es auf Vollständigkeit und Fehler. Das Programm vergleicht die Einträge im Verzeichnis mit den entsprechenden Daten im *FAT*. Findet das Programm Fehler, zeigt es diese durch entsprechende Fehlermeldungen auf dem Bildschirm an. Dies ist z. B. der Fall, wenn eine Datei im Verzeichnis vorhanden ist, der im FAT keine Sektoren zugeordnet sind.

CHKDSK

Mit diesem Dienstprogramm sollten Sie in regelmäßigen Abständen jede Platte überprüfen. Sind Fehler vorhanden, werden Sie darauf aufmerksam gemacht und können diese anschließend vom Programm beheben lassen.

Geben Sie dazu das Programm erneut ein, diesmal mit dem Zusatz /F. Durch diesen Zusatz wird das Programm angewiesen, die Fehler sofort zu beheben. Geben Sie den Zusatz /V mit ein, zeigt das Programm laufend die gerade überprüfte Datei mit den eventuellen Fehlermeldungen an.

Achten Sie darauf, daß Sie das Programm CHKDSK nur unter der Version 3 aufrufen. Für andere Versionen des Betriebssystemes sind eigene Fassungen des Programmes vorhanden. Benutzen Sie die falschen Versionen, ist es möglich, daß wertvolle Daten zerstört werden.

Rufen Sie das Programm ohne weitere Zusätze auf, wird das momentane Laufwerk überprüft. Geben Sie ein

```
CHKDSK
```

und es erscheint nach ein paar Sekunden die Statusmeldung, die folgendermaßen aussehen könnte:

```
Volume ZENITH 001 created JUL 29, 1984 12:09a

  8015872 bytes total disk space
    49152 bytes in 5 hidden files
    28672 bytes in 6 directories
  2830336 bytes in 222 user files
     4048 bytes in 8 bad sectors
     6656 bytes in 1 recovered files
    17280 bytes would be in 2 recovered files
  5079728 bytes available on disk

   327680 bytes total memory
   246992 bytes free
```

Die Statusmeldung enthält folgende Informationen:

- die Gesamtspeicherkapazität der Platte
- die Anzahl und Größe der verdeckten Dateien
- die Anzahl und Größe der Verzeichnisse (wurden keine Subdir angelegt, erscheint diese Meldung nicht)
- die Anzahl und Größe der gespeicherten Dateien
- die Anzahl und Größe der fehlerhaften Sektoren (wurden keine gefunden, erscheint diese Meldung nicht)
- die Anzahl und Größe der durch CHKDSK erneuerten Dateien (wurden keine Dateien erneuert, erscheint diese Meldung nicht)
- die Anzahl und Größe der durch CHKDSK zu erneuernden Dateien (falls fehlerhafte Dateien gefunden wurden und das Programm ohne den Zusatz /F aufgerufen wurde.)
- die noch verbleibende freie Speicherkapazität der Platte
- die Gesamtgröße des Arbeitsspeichers
- die noch verbleibende freie Kapazität des Arbeitsspeichers

Beispiele

Geben Sie nach dem Befehlsnamen einen Dateinamen ein, wird zusätzlich noch einmal nur diese Datei überprüft. Ein Beispiel:

```
CHKDSK WS.COM
```

Jetzt wird nach Überprüfung der Platte die Datei WS.COM überprüft. Wurden keine Fehler gefunden, erscheint erst die Statusmeldung und dann für die Datei WS.COM die Meldung:

```
C:WS.COM
  All specified file(s) are contiguous
```

Diese Mitteilung besagt, daß alle Sektoren der Datei fortlaufend auf der Platte gespeichert wurden.

Dies ist nicht der Fall, z. B. wenn die Datei an einer Stelle gespeichert wird, an der schon eine andere, nicht mehr benötigte, Datei war. Dann kann es sein, daß die Datei immer in die ersten freien Sektoren geschrieben wird. Sie ist dann nicht mehr zusammenhängend, und es erscheint die Meldung:

```
C:WS.COM
   Contains X non-contiguous blocks
```

X bezeichnet die Anzahl der nicht zusammenhängend gespeicherten Gruppen (Gruppe = mehrere Sektoren).

Findet das Programm CHKDSK Fehler auf der Platte, werden diese mit folgendem Kommentar angezeigt:

Errors found, F parameter not specified. Corrections will not be written to disk.	Fehler gefunden, Zusatz /F nicht eingegeben. Berichtigungen werden nicht auf die Platte geschrieben.

Jetzt folgt eine genaue Aufzählung der festgestellten Fehler (s. a. Abschnitt Fehlermeldungen). Die meisten dieser Fehler können vom Programm automatisch korrigiert werden, geben Sie den Zusatz /F ein. Haben Sie beim ersten Prüflauf der Platte den Zusatz nicht eingegeben, erscheint die o. a. Meldung. Sie sollten immer so verfahren, um eine Fehlerliste zu bekommen (mit Zusatz /F wird sofort korrigiert, ohne Fehlerliste). Zum Schluß der Liste könnte z. B. folgende Frage erscheinen:

```
137 lost clusters found in 13 chains.
Convert lost chains to files (Y/N)?
```

Die *Sektoren*, die keiner im Verzeichnis enthaltenen Datei zuzuordnen sind, können jetzt gelöscht, oder als neue Datei gespeichert werden. Im o. a. Beispiel würden 13 Dateien mit insgesamt 274 Sektoren neu angelegt, geben Sie auf die Frage ein **Y** für ja ein. Diese Frage müssen Sie mit **Y** oder **N** beantworten, damit das Programm weiter abgearbeitet werden kann. Da der Zusatz /F nicht eingegeben wurde, erfolgt noch keine Korrektur auf der Platte selber (die Korrektur erfolgt nur bei Eingabe von /F).

Sie sollten hier immer mit **Y** antworten, um die Daten zu speichern. Sie können die Dateien immer noch mit dem Befehl DEL löschen. So haben Sie die Möglichkeit, eventuell einen Teil der fehlerhaften Daten zu behalten.

CHKDSK

Haben Sie die letzte Frage mit **Y** für Ja beantwortet, werden diese Dateien von CHKDSK im Rootdir der Platte angelegt. Die Dateien werden mit FILEnnnn.CHK benannt. *nnnn* ist die fortlaufende Numerierung beginnend ab 0000. Anschließend erscheint die Meldung:

140288 bytes disk space freed	140288 Bytes sind wieder erreichbar

An diese Meldung schließt sich die Statusmeldung für die Platte an, und das Programm ist beendet. Haben Sie die Frage mit **N**(ein) beantwortet, würden keine Dateien angelegt und es erscheint die Meldung:

140288 bytes disk space would be freed	140288 Bytes würden wieder erreichbar sein

Fehlermeldungen

Die folgenden Fehler werden automatisch korrigiert, falls Sie den Zusatz /F eingegeben haben.

Allocation error, size adjusted	Zuordnungsfehler, Größe korrigiert

ERKLÄRUNG:
Es wurde ein Fehler im FAT festgestellt, der von CHKDSK automatisch korrigiert wird.

Entry has a bad link	Eintrag hat eine falsche Verbindung
Entry has a bad size	Eintrag hat eine falsche Größe
Entry has a bad attribute	Eintrag hat ein falsches Merkmal

ERKLÄRUNG:
Jede dieser Meldungen wird durch Fehler im FAT hervorgerufen. Es handelt sich dabei meistens um falsche Sektorenbezeichnungen.

First cluster number is invalid, entry truncated	Erste Gruppennummer ist falsch, der Eintrag ist verstümmelt

ERKLÄRUNG:
Die erste Gruppennummer der Datei weist auf einen falschen Sektor. Die Datei ist nicht mehr zu lesen.

Has invalid cluster, file truncated	Die Datei hat eine falsche Gruppennummer und ist verstümmelt

ERKLÄRUNG:
Im FAT ist ein Sektor der Datei nicht richtig bezeichnet. Die Datei kann nicht mehr bis zum Ende gelesen werden.

Die folgenden Fehler werden vom Programm CHKDSK nicht mehr automatisch durch den Zusatz /F korrigiert.

Cannot CHDIR to root Processing cannot continue	Das Verzeichnis kann nicht überprüft werden. Das Programm wird abgebrochen

ERKLÄRUNG:
Die Diskette ist beschädigt. Starten Sie das System neu, und überprüfen Sie die Diskette mit dem Programm RECOVER.

Cannot CHDIR to filename Tree past this point not processed	Das Verzeichnis kann ab Dateiname nicht weiter überprüft werden Das Programm wird abgebrochen

ERKLÄRUNG:
CHKDSK kann ab der durch Dateiname bezeichneten Datei das Inhaltsverzeichnis nicht weiter überprüfen. Überprüfen Sie die Diskette mit dem Programm RECOVER.

Directory is totally empty, Cannot recover	Inhaltsverzeichnis ist leer, es kann nicht erneuert werden

ERKLÄRUNG:
Das Inhaltsverzeichnis ist fehlerhaft und kann nicht wieder erneuert werden. Formatieren Sie die Diskette neu.

CHKDSK

Disk error reading FAT	Lesefehler des FAT

ERKLÄRUNG:
Das FAT ist beschädigt. Überprüfen Sie die Diskette mit dem Programm RECOVER.

Disk error writing FAT	Schreibfehler des FAT

ERKLÄRUNG:
Das FAT ist beschädigt. CHKDSK kann keine Korrekturen vornehmen. Überprüfen Sie die Diskette mit dem Programm RECOVER.

Disk error writing FAT X	Schreibfehler im FAT X

ERKLÄRUNG:
Das Programm CHKDSK versucht einen Fehler zu beheben. Das FAT ist beschädigt. *X* ist 1 oder 2 und zeigt das beschädigte FAT an. Sind beide Tabellen fehlerhaft, kann die Diskette nicht mehr benutzt werden.

File allocation table bad	Die Dateizuordnungstabelle ist fehlerhaft

ERKLÄRUNG:
Die Diskette ist beschädigt. Starten Sie das System neu, und überprüfen Sie die Diskette mit dem Programm RECOVER.

Filename is cross linked on cluster X	Die Datei ist ab der Gruppennummer X verbunden

ERKLÄRUNG:
Diese Meldung muß zweimal erscheinen, für jede Datei. Korrigieren Sie den Fehler folgendermaßen:

1. Kopieren Sie beide bezeichneten Dateien auf eine andere Diskette.

2. Löschen Sie beide Originaldateien.

3. Überprüfen Sie die beiden Kopien.

Incorrect Dos version	Falsche DOS-Nummer

ERKLÄRUNG:
Sie benutzen das Programm CHKDSK unter einer DOS-Version, die nicht mit der Programmversion übereinstimmt.

Insufficient memory Processing cannot continue	Der Arbeitsspeicher ist zu klein Das Programm wird abgebrochen

ERKLÄRUNG:
Zum Überprüfen dieser Platte ist der Arbeitsspeicher zu klein. Das Programm kann nicht korrekt arbeiten.

Insufficient room in root directory Erase files in root and repeat CHKDSK	Nicht genug Speicherplatz im Inhaltsverzeichnis Löschen Sie Dateien und wiederholen Sie CHKDSK

ERKLÄRUNG:
Im Inhaltsverzeichnis ist kein Platz mehr, um die erneuerten Dateien von CHKDSK einzutragen. Löschen Sie nicht benötigte Dateien, und wiederholen Sie das Programm.

Invalid drive specification	Falsche Laufwerkbezeichnung

ERKLÄRUNG:
Der eingegebene Laufwerkname ist nicht vorhanden oder die Schreibweise war falsch.

Invalid current directory Processing cannot continue	Falsches Verzeichnis Programm wird abgebrochen

ERKLÄRUNG:
Starten Sie das System neu, und wiederholen Sie das Programm CHKDSK.

CHKDSK

Invalid parameter	Falsche Zusatzeingabe

ERKLÄRUNG:
Einer der eingegebenen Zusätze der Befehlszeile war falsch.

Invalid subdirectory entry	Falscher Eintrag im Verzeichnis

ERKLÄRUNG:
Das Programm hat fehlerhafte Informationen im benannten Verzeichnis gefunden. CHKDSK versucht den Fehler zu beheben. Wiederholen Sie das Programm mit dem Zusatz /V.

Probable non-DOS disk Continue (Y/N)?	Keine DOS-Diskette Soll weiter überprüft werden?

ERKLÄRUNG:
Die überprüfte Diskette ist vielleicht nicht im DOS-Format. Geben Sie den Zusatz /F nicht ein, und Sie antworten jetzt mit **Y** für Ja, zeigt das Programm die Fehler und mögliche Korrekturen an. Es wird jedoch nichts auf die Platte geschrieben. Haben Sie den Zusatz /F eingegeben und geben ein **Y** ein, werden die Korrekturen auf die Platte geschrieben.

Unrecoverable error in directory Convert directory to file (Y/N)?	Nicht behebbarer Fehler im Verzeichnis Soll das Verzeichnis in eine Datei eingetragen werden (J/N)?

ERKLÄRUNG:
Antworten Sie mit **Y** für Ja, können Sie anschließend die Daten mit dem Programm DEBUG überprüfen. Antworten Sie mit **N**(ein), werden die Daten gelöscht.

Cannot CHKDSK a Network drive	Ein Laufwerk des Netzwerkes kann nicht überprüft werden

ERKLÄRUNG:
Sie haben versucht, mit dem Befehl CHKDSK ein Laufwerk des Netzwerkes zu überprüfen. Dies ist nicht möglich.

CIPHER Extern

Zweck

verschlüsselt Dateien mit einem gegebenen Wort zu Sicherheitszwecken und entschlüsselt diese Dateien wieder

Syntax

[d:] [Weg] CIPHER Wort > Dateibez
[d:] [Weg] CIPHER Wort < Dateibez
[d:] [Weg] CIPHER Wort < Dateibez1 > Dateibez2

dabei bedeutet:

d:	vor dem Programmnamen bestimmt das aufzurufende Laufwerk
Weg	vor dem Programmnamen bestimmt das aufzurufende Verzeichnis
Wort	Eingabe zum ver- und entschlüsseln der Datei
Dateibez	Datei zum ver- oder entschlüsseln
Dateibez 1	verschlüsselte Datei, die entschlüsselt werden soll
Dateibez 2	neue Dateibezeichnung der entschlüsselten Datei
< ; >	Zeichen zur Datenumleitung

Bemerkungen

Das Programm CIPHER kann im *Netzwerk* **ohne** Einschränkungen benutzt werden.

Mit diesem *Filter* können Sie Ihre Programme und Daten vor unberechtigtem Zugriff schützen. Der Filter *verschlüsselt* die Daten, so daß sie nicht mehr gelesen werden können. Verschlüsselte Programme können nicht mehr abgearbeitet werden.

CIPHER

Die Daten werden mit dem von Ihnen eingegebenen *Wort* verschlüsselt. Dies können Anwenderprogramme oder Daten jeglicher Art sein. Nur mit demselben Wort können Sie die Originaldaten wieder zurückbekommen.

Als *Schlüsselwort* können Sie jede beliebige Zeichenfolge eingeben. Es dürfen maximal soviel Zeichen eingegeben werden, daß die gesamte Befehlszeile nicht mehr als 127 Zeichen umfaßt. Das Wort muß direkt nach dem Befehlsnamen CIPHER eingegeben werden, getrennt durch ein Leerzeichen. Alle folgenden Zeichen bis zum nächsten Leerzeichen bilden das Schlüsselwort. Das Wort selber darf also keine Leerzeichen enthalten. Das Wort kann mit Groß- oder Kleinbuchstaben eingegeben werden. Kleinbuchstaben werden nicht umgeformt. Diese Zeichen müssen dann zum Entschlüsseln ebenfalls wieder als Kleinbuchstaben eingegeben werden.

Geben Sie zum Entschlüsseln der Daten das Wort in genau der selben Reihenfolge wieder ein, wie Sie es zum Verschlüsseln eingegeben haben. Nur so kann der Befehl korrekt abgearbeitet werden.

Die Funktionen der Datenumleitung können Sie zusammen mit dem Filter CIPHER sehr gut gebrauchen. (Datenumleitung wird im Kapitel 8 ausführlich erklärt.) Der Filter liest die zu verschlüsselnden Dateien und gibt sie anschließend verschlüsselt (gefiltert) wieder aus bzw. umgekehrt. Mit dem Zeichen „ > “ werden Dateien ausgegeben, mit dem Zeichen „ < “ werden Dateien eingegeben.

Verschlüsseln einer Datei

Haben Sie Daten im ASCII-Format erfaßt, und wollen Sie diese Daten gegen unberechtigten Zugriff schützen, verschlüsseln Sie diese Daten mit dem Filter CIPHER. Anschließend löschen Sie die Originaldatei. Jetzt können die verschlüsselten Daten nur wieder lesbar gemacht werden, wenn Sie das richtige Schlüsselwort eingeben.

Die zu verschlüsselnde Datei wird WTRGATE.BRK genannt, das Schlüsselwort ist MWMS, und die neue Datei wird SICHER.DAT genannt. Geben Sie ein:

```
CIPHER MWMS < WTRGATE.BRK > SICHER.DAT
```

Nach einigen Sekunden erscheint wieder das Prompt und Ihre Daten sind verschlüsselt. Überprüfen Sie beide Dateien mit dem Befehl TYPE, und Sie sehen den Unterschied. Löschen Sie jetzt die Originaldatei, so sind die Daten vor jedem Zugriff sicher.

Erstellen einer verschlüsselten Datei

Sie können eine Datei auch verschlüsselt erstellen. Dies gilt für Dateien, die Sie direkt von der Tastatur aus eingeben. Rufen Sie CIPHER folgendermaßen auf:

```
CIPHER Wort > Dateibez
```

dabei bedeutet:

Wort	Eingabe zum Verschlüsseln der Datei
Dateibez	Datei zum verschlüsseln

Die so erstellte Datei wird sofort verschlüsselt auf der Diskette gespeichert. Sie müssen dadurch die Datei nicht erst mit einem Texteditor erstellen und anschließend verschlüsseln. Ein Beispiel:

```
CIPHER Tag > B:SECRET.BIN
```

Geben Sie den Text der Datei wie gewohnt über die Tastatur ein. Sie sehen dabei alle Eingaben auf dem Bildschirm. Beenden Sie die Eingabe mit **CTRL-Z.** Jetzt wird der Text verschlüsselt und anschließend als Datei SECRET.BIN im Laufwerk B gespeichert. Das Schlüsselwort ist *Tag.* Sie müssen es zum Entschlüsseln genau so wieder eingeben.

Verschlüsselte Dateien können wie gewohnt mit dem Befehl COPY kopiert werden.

Entschlüsseln einer Datei

Verschlüsselte Dateien können zur Anzeige auf dem Bildschirm entschlüsselt werden. Genau so ist es auch möglich, die Daten in eine andere Datei zu schreiben, mit der dann weitergearbeitet werden kann. Entschlüsseln Sie die im letzten Beispiel erstellte Datei SECRET.BIN durch Eingabe von:

```
CIPHER Tag < SECRET.BIN
```

Der Text wird jetzt entschlüsselt auf dem Bildschirm angezeigt. Die Datei verändern Sie auf der Platte nicht.

Soll der verschlüsselte Text vom Filter gelesen und anschließend entschlüsselt gespeichert werden, leiten Sie die Ausgabe in eine neue Datei um. Der Text wird dann auf dem Bildschirm nicht angezeigt. Entschlüsseln Sie die Datei SECRET.BIN, und schreiben Sie die Daten in die Datei LESBAR.TXT. Das Schlüsselwort muß exakt so eingegeben werden wie zu Anfang. Geben Sie ein:

```
CIPHER Tag < SECRET.BIN > LESBAR.TXT
```

Es existieren jetzt zwei Dateien mit den Daten. Die Datei LESBAR.TXT mit Daten, die Sie weiterverarbeiten können, und die Datei SECRET.BIN mit verschlüsselten Daten, die niemand lesen kann.

Rufen Sie den Filter CIPHER ohne ein entsprechendes Schlüsselwort auf, erscheint die Fehlermeldung:

Mustspecify a code word	Schlüsselwort nicht benannt

Geben Sie den Befehl mit dem richtigen Schlüsselwort erneut ein.

CLS **Intern**

Zweck

löscht den Bildschirm

Syntax

```
CLS
```

Bemerkungen

Mit diesem Befehl *löschen* Sie den gesamten Bildschirminhalt. Der Cursor wird anschließend in die linke obere Ecke gesetzt.

COMMAND Extern

Zweck

ruft die Datei COMMAND.COM als externen Befehl auf

Syntax

[d:] [Weg] COMMAND [D:] [Weg] [cttydev] [/X]

dabei bedeutet:

d: vor dem Programmnamen bestimmt das aufzurufende Laufwerk

Weg vor dem Programmnamen bestimmt das aufzurufende Verzeichnis

D: Name des Laufwerkes, wo der Gerätetreiber gespeichert ist

Weg mehrere Verzeichnisse in Folge in der Form:

[\] [Verzeichnis [\Verzeichnis...]]

cttydev Name des Gerätetreibers

/X Kombination der folgenden Zusatzeingaben:

- **/D** fragt beim Aufruf nicht nach Datum und Zeit
- **/P** definiert COMMAND.COM als permanente Datei, die durch andere Programme nicht mehr überschrieben wird
- **/C** definiert eine Zeichenkette bis zur maximalen Eingabe von 127 Zeichen in der Form: /C Zeichenkette
- **/F** kehrt nach einem Plattenfehler zum Anwenderprogramm zurück, ohne eine Fehlerroutine aufzurufen

Bemerkungen

Das Programm COMMAND kann im *Netzwerk* **ohne** Einschränkungen benutzt werden.

Arbeiten Sie mit der Datei COMMAND.COM als externem Befehl, müssen Sie über den Aufbau von MS-DOS genaue Kenntnisse besitzen. Erste Begriffe werden im Kapitel 9 erklärt.

Die Datei COMMAND.COM ist der *Befehlsinterpreter* des Betriebssystemes. Die Datei wird im Inhaltsverzeichnis offen angezeigt und kann jederzeit gelöscht oder weiterkopiert werden. Dabei muß die Datei nicht an einer bestimmten Stelle der Platte vorhanden sein.

Die Datei COMMAND.COM stellt die *Verbindung* zwischen dem Benutzer und dem Gerät her. Die Datei nimmt die Befehle des Benutzers entgegen und interpretiert diese. Sie wird daher auch als Befehlsinterpreter bezeichnet. Ab der Version 2.0 ist es möglich, diese Datei durch einen anderen Interpreter zu ersetzen. Dazu muß die Datei CONFIG.SYS einen entsprechenden Befehl enthalten.

Die Datei COMMAND.COM besteht aus drei Teilen:

- dem Initialisierungsteil
- dem beständigen Teil
- dem überschreibbaren Teil

Der Initialisierungsteil

Dieser Teil wird oberhalb des nicht überschreibbaren Teiles in den Arbeitsspeicher geladen. Er übernimmt die Kontrolle während des *Systemstartes*. Dieser Teil sucht die niedrigste freie Speicheradresse, ab dem ein Anwenderprogramm geladen werden kann. Hat er diese Funktion erfüllt, wird er nicht mehr benötigt. Das erste zu ladende Programm überschreibt diesen Teil.

Der beständige Teil

Dieser Teil der Datei COMMAND.COM wird beim Systemstart in den Arbeitsspeicher geladen und bleibt dort so lange, bis das System neu gestartet wird. Er enthält die nötigen Programmteile, die erforderlich sind, um Programme zu laden. Er bearbeitet Fehler, die von den Peripheriegeräten verursacht werden. Hier wird ebenfalls die Programmunterbrechung durch die Tastenkombination **CTRL-BREAK** bearbeitet. Ebenso enthält dieser Teil ein Programm zum Nachladen des überschreibbaren Teiles.

Der überschreibbare Teil

Dieser Teil der Datei COMMAND.COM enthält alle internen Befehle. Er gibt das Systemanforderungszeichen aus. Hier befindet sich ebenfalls der Befehlsinterpreter und ein Verarbeitungsprogramm für Batchdateien. Dieser Teil wird an das obere Ende des verfügbaren Speichers geladen.

Sollte ein externer Befehl oder ein Anwenderprogramm viel Arbeitsspeicher benötigen, kann dieser Teil der Datei COMMAND.COM überschrieben werden. Nach Rückkehr zum Betriebssystem wird dieser Teil dann automatisch wieder nachgeladen. Der überschreibbare Teil der Datei COMMAND.COM kann überschrieben werden, weil die dort vorhandenen Funktionen während der Abarbeitung eines anderen Programmes nicht benötigt werden. Diese Programme sind so geschrieben, daß sie alleine ablaufen können. Sie überschreiben diesen Teil erst dann, wenn der restliche Arbeitsspeicher schon belegt ist und zur Ausführung des Programmes weitere Speicherplätze belegt werden müssen.

Befehlseingabe

Mit dem Befehl

```
COMMAND
```

lädt sich die Datei COMMAND.COM selbst in den Arbeitsspeicher. Es erscheinen noch einmal die Abfragen für Datum und Zeit. Geben Sie den Zusatz /D mit ein, erfolgt die Abfrage nicht. Durch den Zusatz /P wird die Datei permament in den Speicher geladen. Sie wird dann nicht mehr von Anwenderprogrammen überschrieben, wenn der Arbeitsspeicher zu klein geworden ist.

/C - Zeichenkette

Geben Sie den Zusatz /C *Zeichenkette* mit ein, muß dieser Zusatz als letzte Eingabe der Befehlszeile erfolgen. Alle Zeichen nach dem /C werden als Bestandteil der Zeichenkette interpretiert. Die Datei COMMAND.COM sucht die Zeichenkette nach einem Befehlsnamen oder einer anderen ausführbaren Datei ab und führt dann den Befehl aus.

Die Datei COMMAND.COM wird z. B. als ausführbare Datei vom Programm APPLY benötigt. Das Programm APPLY führt als selbständiges Programm einen internen Befehl mehrere Male aus. Da es den überschreibbaren Teil des Befehlsinterpreters (die Datei COMMAND.COM) überschrieben hat, ist es normalerweise nicht möglich, interne Befehle auszuführen.

Dadurch, daß die Datei COMMAND.COM unter der Kontrolle von APPLY geladen wird, können dann interne Befehle ausgeführt werden (s. a. Befehlserklärung APPLY). Andere Anwendungsprogramme benutzen diese Möglichkeit ebenfalls, z. B. der BASIC-Interpreter.

Der Zusatz /C kann nicht gleichzeitig mit dem Zusatz /P eingegeben werden.

/F - keine Fehlerroutine

Geben Sie den Zusatz /F ein, werden *Plattenfehler* nicht mehr bearbeitet. Das Betriebssystem kehrt sofort wieder in das Anwenderprogramm zurück. Normalerweise erscheint bei einem Plattenfehler die genaue Fehlerursache mit der Aufforderung:

Abort, Retry, Ignore?	Abbrechen, Wiederholen, Ignorieren?

Durch Eingabe eines entsprechenden Buchstabens bestimmen Sie den weiteren Programmverlauf. Geben Sie ein A für Abort (Abbrechen) ein, wird damit ein Anwenderprogramm sofort beendet. Bei Aufruf der Datei COMMAND.COM mit dem Zusatz /F geschieht dies nicht.

COMMAND

Sie können für diesen Zusatz einen weiteren Befehl der Datei benutzen. Rufen Sie die Datei mit dem Zusatz /F auf. Geben Sie ein:

```
COMMAND /F
```

Rufen Sie danach das Anwenderprogramm auf, das auch bei einem Plattenfehler nicht verlassen werden sollte. Haben Sie das Programm beendet, kehren Sie wie gewohnt in das Betriebssystem zurück. Sie befinden sich dann aber noch in der nachgeladenen Datei COMMAND.COM. Geben Sie jetzt den Befehl

```
EXIT
```

ein, verlassen Sie diese Datei wieder und sind im ursprünglichen Befehlsinterpreter von MS-DOS.

Fehlermeldungen

Bad call format — Falscher Aufruf

ERKLÄRUNG:
Die einem Gerätetreiber (Programm zum Anschluß eines Peripheriegerätes) übergebenen Variablen sind falsch. Benutzen Sie einen anderen Gerätetreiber, oder beheben Sie (lassen Sie vom Verkäufer beheben) den Programmfehler des Treibers. Ist dies ein Treiber von Zenith Data Systems, wenden Sie sich an Ihren zuständigen Händler.

Bad command or filename — Befehl oder Datei falsch

ERKLÄRUNG:
Der eingegebene Befehl wurde falsch geschrieben, oder die Datei ist auf dem Laufwerk nicht vorhanden.

Bad or missing Command Interpreter	Datei COMMAND.COM fehlt oder ist falsch

ERKLÄRUNG:
Die aufgerufene Datei COMMAND.COM ist nicht auf dem Arbeitslaufwerk vorhanden.

Error writing to device	Schreibfehler des Peripheriegerätes

ERKLÄRUNG:
Das Peripheriegerät, das Sie mit dem Zusatz cttydev benannt haben, kann keine Daten empfangen.

EXEC failure	Fehler durch Systemanruf

ERKLÄRUNG:
Diese Fehlermeldung kann durch eine der folgenden Ursachen hervorgerufen werden:

- Die aufgerufene Datei existiert nicht in der Form, in der sie aufgerufen wurde.
- Das aufgerufene .EXE-Programm enthält im Header nicht die von MS-DOS geforderten Informationen.
- Der Arbeitsspeicher ist zu klein.
- Durch die Datei CONFIG.SYS wurden Systemerweiterungsprogramme geladen. Deshalb ist der Arbeitsspeicher zu klein.
- Es sind Anpassungsprogramme zusätzlich geladen worden. Der Arbeitsspeicher ist zu klein.
- Es wurde ein falscher Funktionsaufruf benutzt. Funktion Nummer 0, 1 oder 3 ist richtig.

COMMAND

Invalid device	Falsches Peripheriegerät

ERKLÄRUNG:
Sie wollen mit dem Zusatz /cttydev auf ein Peripheriegerät zugreifen, das von MS-DOS nicht unterstützt wird, oder das bezeichnete Gerät ist nicht angeschlossen.

Invalid directory	Falsches Inhaltsverzeichnis

ERKLÄRUNG:
Das angegebene Verzeichnis ist nicht vorhanden.

Invalid drive specification	Falsche Laufwerkbezeichnung

ERKLÄRUNG:
Der eingegebene Laufwerkname ist nicht vorhanden, oder die Schreibweise war falsch.

Invalid number of parameters	Falsche Anzahl der Zusatzeingaben

ERKLÄRUNG:
Sie haben zu viele oder zu wenige Zusatzeingaben vorgenommen. Überprüfen Sie Ihre Eingabe.

Invalid parameter	Falsche Variable

ERKLÄRUNG:
Eine der eingegebenen Variablen der Befehlszeile war falsch.

Invalid path or file name	Weg oder Dateiname falsch

ERKLÄRUNG:
Das eingegebene Verzeichnis oder der Dateiname war falsch.

COMP **Extern**

Zweck

vergleicht den Inhalt von zwei Dateien oder zwei Gruppen von Dateien auf Gleichheit

Syntax

[d:] [Weg] COMP ?
[d:] [Weg] COMP [Dateibez1 [Dateibez2]]

dabei bedeutet:

d:	vor dem Programmnamen bestimmt das aufzurufende Laufwerk
Weg	vor dem Programmnamen bestimmt das aufzurufende Verzeichnis
?	zeigt eine Erklärung des Programmes
Dateibez1	erste zu vergleichende Datei oder Dateigruppe
Dateibez2	zweite zu vergleichende Datei oder Dateigruppe

Bemerkungen

Das Programm COMP kann im *Netzwerk* **ohne** Einschränkungen benutzt werden.

Benutzen Sie das Programm COMP, um nach einem Kopiervorgang zu prüfen, ob die einzelne Datei richtig kopiert wurde. Haben Sie eine Diskette komplett kopiert, überprüfen Sie diese Diskette schneller mit dem Befehl DISKCOMP. Die zu vergleichenden Dateien können dabei auf verschiedenen Laufwerken oder in verschiedenen Verzeichnissen gespeichert sein. Beide Dateien müssen immer gleich groß sein.

COMP

Werden vom Programm COMP Differenzen gefunden, erfolgt die Ausgabe mit hexadezimalen Werten in der Form:

```
Compare error at offset xxxxxxxx
File 1 = yy
File 2 = zz
```

dabei bedeutet:

xxxxxxxx	Adresse in der Datei, die ungleiche Informationen enthält
yy	Wert der Adresse xxxxxxxx in der Datei 1
zz	Wert der Adresse xxxxxxxx in der Datei 2

Diese Meldung wird für jede Differenz einzeln ausgegeben. Findet COMP mehr als 10 Unterschiede, wird die Überprüfung abgebrochen. Die Dateien sind dann zu unterschiedlich und sollten noch einmal kopiert werden. In diesem Fall erscheint die Meldung:

10 Mismatches - ending compare	10 Unterschiede – Der Vergleich wird abgebrochen

Zu Anfang des Vergleich wird das letzte Byte jeder Datei mit dem Zeichen für Dateiende, dies ist ein CTRL-Z (hexadezimaler Wert 1A), verglichen. Findet das Programm COMP diesen Wert, wird mit der Überprüfung begonnen. Ist das Byte nicht vorhanden, erscheint folgende Meldung:

EOF mark not found	Das Zeichen für Dateiende ist nicht vorhanden

Das Dateiende wird überprüft, weil verschiedene Programme ihre Dateien immer in Blöcken von 128 Bytes speichern. Diese Programme sind aber in Wirklichkeit geringfügig kleiner, als im Inhaltsverzeichnis angezeigt. Das Dientprogramm COMP vergleicht Dateien immer auf Grund ihrer Länge im Verzeichniseintrag. Fehlt das Zeichen für Dateiende, werden die Dateien bis zum Schluß verglichen und damit auch die Füllbytes bis zum Ende des 128-Byte-Block. Diese Bytes werden fast immer unterschiedlich sein, das Programm COMP zeigt dann entsprechende Fehlermeldungen. In diesem Falle können Sie die Fehlermeldungen jedoch ignorieren.

Haben Sie das Programm COMP zusammen mit den Namen für die zu vergleichenden Dateien aufgerufen, beginnt der Vergleich sofort. Sie haben dann keine Gelegenheit mehr, eine Diskette zu tauschen. Bei Geräten mit nur einem Laufwerk sollten Sie das Programm nur mit seinem Namen aufrufen. Sie werden dann nach den zu vergleichenden Dateien gefragt. Das Programm startet erst nach Eingabe des zweiten Namens.

Programmaufruf

Sie können das Programm COMP mit verschiedenen Befehlen aufrufen. Mit Eingabe des Fragezeichens rufen Sie eine kurze Erklärung auf. Geben Sie ein

```
COMP ?
```

und es erscheint folgende Übersicht:

```
                    COMP Version 3.01
Copyright (C) 1985 Zenith Data Systems Corporation

The COMP utility compares the contents of two files or two specified sets of files.
Normally, COMP is used after a copy operation to verify that the files were copied
correctly. The files that are specified for comparison can be on the same drive or on
different drives. They can also be in the same or different directories. Wildcard
characters are allowed in both file specifications.

Syntax: COMP ?
        COMP [filespec1 [filespec2]]
```

Für eine Befehlseingabe ist das Programm erneut aufzurufen. Dabei können Sie zwischen zwei Möglichkeiten wählen:

- Sie geben die Dateinamen in der Befehlszeile ein
- Sie geben die Dateinamen auf eine entsprechende Anfrage ein

COMP

Rufen Sie das Programm mit seinem Namen ohne weitere Zusätze auf, geben Sie ein

COMP

und es erscheint die Aufforderung, den ersten Dateinamen einzugeben:

Enter primary file name:_	Erster Dateiname:_

Geben Sie hier den Namen der ersten zu vergleichenden Datei ein. Befindet sich die Datei nicht im aktuellen Verzeichnis, kann dem Dateinamen ein Verzeichnisname und, falls erforderlich, auch eine Laufwerkbezeichnung vorangestellt werden. Beenden Sie die Eingabe durch Betätigen der Taste **RETURN**. Es erscheint danach folgender Text:

Enter 2nd file name or drive id:_	Zweiter Dateiname oder Laufwerkbezeichnung:_

Sollen zwei Dateien mit gleichem Namen verglichen werden, ist jetzt nur die Eingabe der Laufwerkbezeichnung und/oder ein Verzeichnisname erforderlich. (Sie können jedoch auch den kompletten Namen eingeben.) Beenden Sie auch diese Eingabe mit **RETURN**. Das Programm beginnt dann mit dem Vergleich.

Für beide Dateinamen ist die Eingabe von Globalzeichen (* und ?) möglich. Das Programm COMP vergleicht bei Verwendung dieser Zeichen alle so bezeichneten Dateien.

Findet das Programm unterschiedliche Werte, erscheint für jede Datei auf dem Bildschirm eine entsprechende Meldung. Ist der Vergleich beendet, und es wurden keine Unterschiede festgestellt, erscheint die Meldung

Files compare OK	Beide Daten sind gleich

und darunter erscheint die Frage:

Compare more files (Y/N)?_	Möchten Sie weitere Dateien vergleichen? (J/N)_

Bei Eingabe eines **Y** für Ja, beginnt das Programm erneut. Geben Sie **N**(ein) ein, verlassen Sie das Programm und es erscheint das Prompt des Betriebssystemes. Es ist nur die Eingabe von **Y** oder **N** möglich. Bei allen anderen Eingaben wird die Frage wiederholt.

Rufen Sie das Programm zusammen mit einem Dateinamen oder einer Laufwerkbezeichnung auf, wird dieser Name vom Programm als erster Dateiname angenommen. Sie werden dann nur noch aufgefordert, den zweiten Dateinamen einzugeben. Geben Sie z. B. ein

COMP A:TEST.DOC

erscheint folgende Meldung:

Enter 2nd file name or drived id:_	Zweiter Dateiname oder Laufwerkbezeichnung:_

Nach Eingabe eines Dateinamens oder einer Laufwerkbezeichnung beginnt das Programm den Vergleich.

Beispiel

Für das folgende Beispiel wird die Datei BRIEF.TXT des Laufwerkes A mit der Datei BRIEF.BAK des Laufwerkes B verglichen. Rufen Sie das Programm mit folgendem Befehl auf:

COMP A:BRIEF.TXT B:BRIEF.BAK

Nach Betätigen der Taste **RETURN** beginnt der Vergleich. Auf dem Bildschirm werden noch einmal die beiden Dateinamen angezeigt.

A:BRIEF .TXT AND B:BRIEF .BAK

Findet das Programm COMP jetzt keine Unterchiede, erscheinen zum Schluß folgende Meldungen:

Files compare OK	Beide Dateien sind gleich
Compare more files (Y/N)?_	Möchten Sie weitere Dateien vergleichen? (J/N)_

Findet das Programm COMP unterschiedliche Werte in den Dateien, könnte folgender Text erscheinen:

Compare error at offset 4B2H	Unterschiedliche Werte bei Adresse 4B2H
File 1 = 20 File 2 = 32	Datei 1 = 20 Datei 2 = 32

Diese Fehlermeldung wird maximal für 10 Adressen ausgegeben. Danach wird der Vergleich mit folgender Meldung abgebrochen:

10 mismatches found - ending compare	Es wurden mehr als 10 Fehler gefunden. Der Vergleich wird abgebrochen.

Sie sollten dann die Datei erneut kopieren.

Fehlermeldungen

EOF mark not found	Das Zeichen für Dateiende ist nicht vorhanden

ERKLÄRUNG:
Das letzte Zeichen der Datei ist kein CTRL-Z. Wird diese Datei vom Programm mit 128-Byte-Blöcken gespeichert, können die letzten Bytes unterschiedliche Werte haben. Diese Bytes gehören nicht mehr zur Datei. Die Unterschiede brauchen nicht beachtet zu werden.

File not found	Datei nicht vorhanden

ERKLÄRUNG:
Das Programm COMP kann die bezeichnete Datei nicht finden. Überprüfen Sie Ihre Eingabe.

Files are different sizes	Die Dateien haben verschiedene Größen

ERKLÄRUNG:
Das Programm vergleicht zuerst die im Verzeichnis eingetragene Dateigröße. Stimmen diese Werte nicht überein, kann der Vergleich nicht durchgeführt werden.

Incorrect DOS-Version	Falsche DOS-Version

ERKLÄRUNG:
Sie benutzen das Programm COMP unter einer DOS-Version, die nicht 3.0 oder höher ist.

Invalid drive specification	Falsche Laufwerkbezeichnung

ERKLÄRUNG:
Sie haben ein nicht vorhandenes Laufwerk bezeichnet. Überprüfen Sie Ihre Eingabe.

10 mismatches found - ending compare	Es wurden mehr als 10 Fehler gefunden. Der Vergleich wird abgebrochen.

ERKLÄRUNG:
Die beiden Dateien sind zu unterschiedlich. Kopieren Sie die Datei neu.

System error: Press CTRL-C	Fehler des Systems. Betätigen Sie CTRL-C.

ERKLÄRUNG:
Das Betriebssystem hat einen Fehler beim Plattenzugriff gemeldet. Beenden Sie das Programm mit der Tastaturkombination CTRL-C. Starten Sie den Vergleich dann erneut. Wird der Fehler noch einmal gemeldet, überprüfen Sie mit dem Befehl CHKDSK die Diskette oder Winchester.

CONFIGUR Extern

Zweck

erstellt das Übertragungsprotokoll für die seriellen und parallelen Ausgänge, bestimmt die Art der Bezeichnung für Winchesterteilbereiche.

Syntax

[d:] [Weg] CONFIGUR

dabei bedeutet:

d: vor dem Programmnamen bestimmt das aufzurufende Laufwerk

Weg vor dem Programmnamen bestimmt das aufzurufende Verzeichnis

Bemerkungen

Das Programm CONFIGUR kann im Netzwerk ohne Einschränkungen benutzt werden.

Mit dem Dienstprogramm CONFIGUR können Sie Ihren Zenith PC an die verschiedenen Peripheriegeräte anpassen. Diese Geräte benötigen ein spezielles *Übertragungsprotokoll,* um die empfangenen Daten richtig zu verarbeiten. Mit Übertragungsprotokoll wird die Art des *Datenaustausches* zwischen einem Computer und einem Peripheriegerät bezeichnet. Sie können an Ihren Computer z. B. verschiedene parallele oder serielle Drucker oder auch Modems anschließen. Jedes Gerät arbeitet anders und benötigt daher ein anderes Protokoll. Mit dem Programm CONFIGUR können Sie sämtliche Ausgänge Ihres Computers den Anforderungen der anzuschließenden Geräte anpassen.

Der erste parallele Ausgang (LPT1) ist bereits für den Anschluß eines parallelen Druckers vorbereitet. Schließen Sie nur einen Paralleldrucker an, brauchen Sie das Programm CONFIGUR nicht aufrufen.

Das Programm ist *menügesteuert.* Das Hauptmenü führt Sie in Teilmenüs, in denen Ihnen jeweils die Auswahlmöglichkeiten angezeigt werden. Nur aus diesen Möglichkeiten können Sie wählen. Die so bestimmten Daten werden dann in der Datei IO.SYS gespeichert. Bevor Sie sich für eine Möglichkeit entscheiden, sollten Sie immer das jeweilige Gerätehandbuch hinzuziehen. Wollen Sie eine angezeigte Möglichkeit auswählen, geben Sie den entsprechenden Buchstaben ein. Andere Eingaben als die gezeigten Buchstaben sind nicht möglich.

Sie können mit dem Programm zwei verschiedene Arten von logischen Ausgängen anpassen:

- parallele Ausgänge
- serielle Ausgänge

Beide Ausgänge werden von unterschiedlichen *Gerätetreibern* (Device Driver) angesteuert. Der Treiber für den parallelen Ausgang ist der Treiber mit dem Namen: PRN (LPT1, LPT2, LPT3). Der Treiber für den seriellen Ausgang ist der Treiber AUX (COM1, COM2). In diesen Treibern werden die von Ihnen ausgewählten Werte abgespeichert. Sie sind dann immer vorhanden, wenn auf dieses Gerät zugegriffen wird.

Tabelle 11.2 Namen der Ausgänge

NAME	AUSGANG
PRN oder LPT1	erster Parallelausgang
AUX oder COM1	erster Seriellausgang
COM2	zweiter Seriellausgang
LPT2	zweiter Parallelausgang
LPT3	dritter Parallelausgang

CONFIGUR

Diese Art der Programmierung erreicht mit wenig Aufwand ein Maximum an Flexibilität. Obwohl an einen logischen Ausgang verschiedene Peripheriegeräte angeschlossen werden können, ist es nicht jedesmal erforderlich, dafür das Betriebssystem neu zu schreiben. Es werden nur die notwendigen Werte der Tabelle im Gerätetreiber geändert. Rufen Sie dann ein Peripheriegerät auf, rufen Sie es mit dem Namen des Gerätetreibers auf (PRN oder AUX). Das Betriebssystem erhält alle notwendigen Informationen aus der Tabelle und kann die Daten korrekt zum Gerät senden.

Rufen Sie das Programm CONFIGUR ein weiteres Mal auf, um den selben Ausgang für ein anderes Gerät vorzubereiten, werden nur die Werte der Tabelle geändert.

In den folgenden Abschnitten werden alle Möglichkeiten der Anpassung ausführlich beschrieben. Im einzelnen werden folgende Programmteile erklärt:

1. Das *Hauptmenü,* es wird von Ihnen mehrfach aufgerufen. Jedesmal werden die verschiedenen Auswahlmöglichkeiten beschrieben.

2. Anpassung eines *Parallelausganges* (LPT), hier wird ein Drucker angeschlossen (für einen seriellen Drucker muß dieser Ausgang umgeleitet werden).

3. Anpassung eines *Seriellausganges* (COM), hier wird ein Modem oder serieller Drucker angeschlossen.

4. Umleitung eines *Parallelausganges,* das wird häufig für einen seriellen Drucker notwendig sein. Es geschieht in folgenden Einzelschritten:

   A. Anpassung eines Seriellausganges für das spezielle Gerät (s. a. unter 3)

   B. Umleitung des Parallelausganges zum Seriellausgang mit der gleichen Nummer (z. B. LPT1 immer zu COM1, nie zu COM2)

   C. Anschluß des Gerätes am Seriellausgang nur mit einem Null-Modem-Kabel. Nehmen Sie kein anderes Kabel.

5. Automatische Bezeichnung der Winchesterteilbereiche, die hier auf manuelle Bezeichnung umgestellt werden.

Das Hauptmenü

Rufen Sie das Programm mit seinem Namen auf:

CONFIGUR

Bevor das Menü erscheint, werden alle angeschlossenen Ausgänge überprüft. Sollten keine vorhanden sein, kann das Programm nicht ablaufen, und es erscheint die Meldung:

No serial or parallel devices are attached. No configuration is possible.	Es ist kein Seriell- oder Parallelausgang vorhanden. Eine Anpassung ist nicht möglich.

Ihr Zenith PC wird standardmäßig mit einem seriellen und einem parallelen Ausgang geliefert (s. a. Bild 11.1). Das Menü sieht dann folgendermaßen aus:

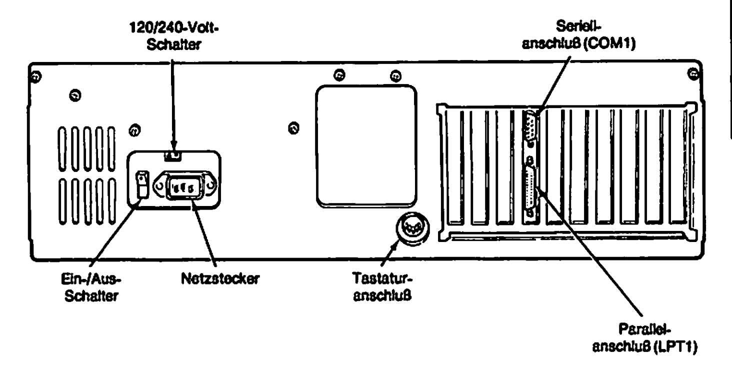


Bild 11.1
Rückseite des PC

CONFIGUR

```
                CONFIGUR Version 3.01
Copyright (C) 1985 Zenith Data Systems Corporation

Use one of the following options to configure a device

A: Configure LPT device
B: Configure COM device
C: Change automatic partition assignment flag

D: Exit with no changes

Enter selection (A-D):_
```

Sie haben jetzt die Möglichkeit, mit einem der Buchstaben **A, B, C** oder **D** Ihre Wahl zu treffen. Sie können mit der Möglichkeit A einen *Parallelausgang* definieren, mit der Möglichkeit B einen *Seriellausgang* definieren, mit der Möglichkeit C die Art der Bezeichnung für Teilbereiche der Winchester ändern und mit der Möglichkeit D das Programm wieder *verlassen.* Während der Anpassung erscheint dieses Menü mehrmals, je nach den Erfordernissen um einige Positionen erweitert. Das Menü könnte dann z. B. so aussehen:

```
                CONFIGUR Version 3.01
Copyright (C) 1985 Zenith Data Systems Corporation

Use one of the following options to configure a device

A: Configure LPT device
B: Configure COM device
C: Change automatic partition assignment flag

Use one of the following to modify an existing system

D: Exit programm
E: Make changes to disk
F: Make changes to memory
G: Make changes to both disk and memory

Enter selection (A-G):_
```

Mit den ersten beiden Möglichkeiten (**A** oder **B**) können Sie wieder einen Ausgang anpassen. Mit **E**, **F** und **G** können Sie die vorgenommenen Änderungen speichern, und mit **D** können Sie das Programm ohne Speicherung der Änderungen verlassen. Geben Sie ein **E** oder **G** ein, werden Sie nach einer Laufwerkbezeichnung gefragt:

Enter drive name with system to modify (A-X):	Geben Sie den Laufwerknamen mit dem zu ändernden System ein (A-X):

Damit haben Sie die Möglichkeit, das Betriebssystem an verschiedene Peripheriegeräte anzupassen. Bei einem Gerätewechsel tauschen Sie die Systemdiskette mit aus und brauchen das Dienstprogramm CONFIGUR nicht neu aufzurufen. Die Änderungen können nur auf nicht schreibgeschützte Disketten geschrieben werden.

Die Möglichkeit **F** erlaubt Ihnen, das momentane Betriebssystem zu ändern, ohne diese Änderung zu speichern. Nach einem neuen Systemstart ist die Änderung nicht mehr vorhanden.

Die Möglichkeiten **A** und **B** leiten Sie in weitere Menüs, mit denen Sie die gewünschten Änderungen vornehmen.

Anpassung eines Parallelausganges

Haben Sie im Hauptmenü ein **A** eingegeben, erscheint jetzt folgendes Bild:

```
Use one of the following options to select the type of configuration

A: Map parallel output to serial output
B: Configure parallel device
C: Change automatic partition assignment flag

D. Exit

Enter selection (A-D):
```

CONFIGUR

Sie können jetzt wählen, ob Sie einen *Parallelausgang* umleiten (Möglichkeit **A**), einen Parallelausgang definieren (Möglichkeit **B**) oder das Programm verlassen wollen (Möglichkeit **D**). Geben Sie ein **D** ein, erscheint wieder das Hauptmenü. Geben Sie zuerst ein **B** ein, um den Ausgang zu definieren. Es erscheint folgendes Bild:

```
Select the parallel device to be configured

A. LPT1

B. Exit

Enter selection (A-B):
```

In diesem Menü werden immer nur die vorhandenen Ausgänge angezeigt (in diesem Fall einer, maximal sind drei möglich). Geben Sie jetzt ein **A** ein, erscheinen nacheinander zwei Fragen:

```
Answer the following questions with Y for Yes and N for No

Strip parity on output? (Y/N) <N>
Map lower case to upper case on output? (Y/N) <N>
```

Sie können das *achte Bit* eines jeden auszusendenden Zeichens auf 0 setzen (erste Frage), und Sie können sämtliche *Kleinbuchstaben* in Großbuchstaben umwandeln lassen (Frage zwei). Beantworten Sie die Fragen mit einem **RETURN,** werden die in Klammern vorgegebenen Werte übernommen, in diesem Falle also Nein.

Nach diesen Fragen erscheinen folgende Fragen:

```
If you do not wish a pad character, simply
 press the RETURN key, and then enter a zero
 as the number of pad characters, otherwise press
 the actual key character you wish to pad.

For example, to pad after all carriage returns, press
 the RETURN key.

Press the key corresponding to your desired pad character:
Enter the number of pad characters to send (0-255):
```

Mit diesen beiden Fragen werden Sie aufgefordert, das Zeichen zu bestimmen, nach dem vom System *Füllzeichen* (Pad Character) gesendet werden sollen. Die Zahl der zu sendenden Zeichen können Sie mit der zweiten Antwort bestimmen.

Diese Möglichkeit ist nur für ältere Drucker gedacht, die nicht sehr schnell arbeiten. Ein langsamer Drucker benötigt sehr viel mehr Zeit, um nach einem Zeilenvorschub wieder zur ersten Spalte zu gelangen. Damit keine Zeichen verlorengehen, werden in dieser Zeit Füllzeichen gesendet. Die Anzahl der Füllzeichen bestimmen Sie mit der zweiten Antwort. Ebenso ist die nächste Frage aus dem gleichen Grunde auch hauptsächlich für ältere Geräte gedacht:

The timeout value is used to give slow devices time to respond to Input/Output requests. A small value is usually sufficient, but a number from 0 to 255 can be entered.

Enter timeout value for LPT1:

Mit diesem Wert können Sie bestimmen, wie lange das System auf eine *Antwort* des Peripheriegerätes wartet, bevor es eine Fehlermeldung ausgibt und den Vorgang abbricht. Diese Möglichkeit ist nicht nur für Drucker gedacht, sondern auch für andere Geräte. Falls erforderlich, reicht hier ein kleiner Wert. Sie können aber eine Zahl von 0 bis 255 eingeben, je nachdem wie langsam das Gerät ist. Die von Zenith Data Systems gelieferten Geräte benötigen diese Wartezeit nicht. Haben Sie hier einen Wert eingegeben (z. B. 0), erscheint wieder das erste Untermenü:

Use one of the following options to select the type of configuration

A: Map parallel output to serial output
B: Configure parallel device

C. Exit

Enter selection (A-C):

CONFIGUR

Haben Sie jetzt alle Änderungen vorgenommen, verlassen Sie mit der Eingabe von C diesen Teil des Programmes und gelangen wieder in das Hauptmenü:

```
                CONFIGUR Version 3.01
Copyright (C) 1985 Zenith Data Systems Corporation

Use one of the following options to configure a device

A: Configure LPT device
B: Configure COM device
C: Change automatic partition assignment flag

Use one of the following to modify an existing system

D: Exit programm
E: Make changes to disk
F: Make changes to memory
G: Make changes to both disk and memory

Enter selection (A-G):_
```

Das Hauptmenü ermöglicht Ihnen, die vorgenommenen Änderungen zu speichern. Wählen Sie **E, F** oder **G.** Danach können Sie das Programm verlassen.

Anpassung eines Seriellausganges

Haben Sie im Hauptmenü ein **B** eingegeben, erscheint jetzt folgendes Bild:

```
Select the serial port to be configured

A. COM1
B. COM2

C. EXIT

Enter selection (A-C):_
```

In diesem Menü werden nur die vorhandenen Ausgänge angezeigt (in diesem Fall zwei). Bestimmen Sie jetzt den anzupassenden *Ausgang* mit der Eingabe des entsprechenden Buchstabens. Es erscheint dann folgendes Bild:

```
A. Compatibility mode (2400 baud, DTR and RTS pos.)
B. MX-80 (4800 baud, DTR pos, (pin 20)
C. H/Z-25 (4800 baud, RTS pos, (pin 4)
D. H-14/WH-24 (4800 baud, RTS Neg, (pin 4))
E. Diablo 630/1640 (1200 baud , ETX/ACK)
F. WH-23/WH-33/WH-43 modem (300 baud, No handshake)
G. WH-12 Votrax Type-N-Talk (4800 baud, RTS Pos. (pin 4))
H. User Defined
I. Exit with no changes

Enter selection (A-I):_
```

Ist Ihr Gerät in dieser Liste aufgeführt, geben Sie den entsprechenden Buchstaben ein und alle Änderungen sind bereits vorgenommen. Die Liste zeigt eine *Auswahl* der gebräuchlichsten Drucker und Modems, sowie einen Modus, der IBM-Kompatibilität für serielle Geräte herstellt (Auswahl **A**). Ist Ihr Gerät nicht aufgeführt, wählen Sie Möglichkeit **H** (User Defined). Diese Möglichkeit wird im folgenden Abschnitt ausführlich beschrieben:

Es erscheinen folgende Fragen:

```
Strip parity on input? (Y/N) <N>
Strip parity on output? (Y/N) <N>
Map lowercase to uppercase on input? (Y/N) <N>
Map lowercase to uppercase on output? (Y/N) <N>
```

Sie können das *Prüfbit* (achtes Bit) auf Null setzen. Dies kann für den Ein- und Ausgang getrennt bestimmt werden (Fragen 1 und 2). Ebenso können Sie für den Ein- und Ausgang getrennt bestimmen, ob alle *Kleinbuchstaben* in Großbuchstaben umgewandelt werden sollen (Frage 3 und 4).

CONFIGUR

Danach erscheint folgendes Menü:

Select one of the following baud rates

A. 110
B. 150
C. 300
D. 600
E. 1200
F. 2400
G. 4800
H. 9600

Enter one of the baud rate values:_

Mit *Baudrate* wird die Übertragungsgeschwindigkeit der gesendeten Zeichen zwischen Computer und Peripheriegerät bezeichnet. Beide Geräte müssen auf die gleiche Baudrate eingestellt sein, damit die Verbindung zu Stande kommt. Den korrekten Wert entnehmen Sie Ihrem Gerätehandbuch. Nach Eingabe der Baudrate erscheint folgende Auswahl:

Use one of the following stop bit values

A. 1 Stop bit
B. 2 Stop bits

Enter one of the stop bit values:_

Mit dem/den *Stopbit* wird das Ende der Datenübertragung gekennzeichnet. Entnehmen Sie auch diesen Wert Ihrem Handbuch. Es erscheint danach die folgende Auswahl:

Use one of the following parity selections

A. No parity
B. Odd parity
C. Even parity

Enter one of the parity values:_

Durch die Art der *Gleichheit* (Parity) überprüft der Computer den Datenaustausch mit dem Peripheriegerät auf Fehler. Wählen Sie auch hier den Ihrem Gerät entsprechenden Wert aus. Nach Eingabe des Wertes erscheint:

```
Use one of the following to select the word length
NOTE: Word length is exclusive of stop bits and parity

A. 7 bit words
B. 8 bit words

Enter one of the word length values:_
```

Mit diesem Wert wird die *Anzahl* der Bits bestimmt, die ein Zeichen bilden. Auch dieser Wert ist bei jedem Gerät unterschiedlich und muß dem Handbuch entnommen werden.

Jetzt erscheint wieder ein Untermenü, das Sie in weitere Menüs führt. Es wird die Art der *Verständigung* zum vorübergehenden Stop des Datentransfers bestimmt (Handshake). Da der Computer immer die Zeichen schneller senden kann, als sie vom Peripheriegerät verarbeitet werden können, ist nach kurzer Zeit der *Pufferspeicher* des Gerätes voll. Dann muß der Computer solange warten, bis im Puffer wieder Platz ist. Es erscheint folgendes Bild:

```
Use the following to select a handshaking protocol

A. No Handshaking
B. ETX/ACK
C. DC1/DC3
D. Compatibility Mode, DTR and RTS Positive
E. RTS Positive (pin 4)
F. RTS Negative (pin 4)
G. DTR Positive (pin 20)
H. DTR Negative (pin 20)

Enter one of the handshake values:
```

Es gibt drei verschiedene Gruppen des Handshakes:

- Kein Handshake (Auswahl A)
- Software Handshake (Auswahl B - C)
- Hardware Handshake (Auswahl D - H)

Die erste Möglichkeit ist für Geräte, die keine besonderen Zeichen zum *Stop* benötigen. Für diese Geräte benötigt der Computer ebenfalls keine Warteschleife. Sie können aber *Füllzeichen* vorgeben.

Die Gruppe der softwaregesteuerten Geräte benötigt Zeichen zum *Stop,* aber keine Warteschleife. Dafür erscheint aber bei Auswahl von **B** (und nur bei B) die zusätzliche Frage:

```
Number of characters between the ETX/ACK handshake (0-255):_
```

Geben Sie die gerätespezifischen Werte jeweils ein.

Die Gruppe der hardwaregesteuerten Geräte erlaubt die Eingabe von *Warteschleifen* und *Füllzeichen* gleichzeitig. Sie werden jeweils nach den einzelnen Werten gefragt:

```
If you do not wish a pad character, simply
 press the RETURN key, and then enter a zero
 as the number of pad characters, otherwise press
 the actual key character you wish to pad.

For example, to pad after all carriage returns, press
 the RETURN key.

Press the key corresponding to your desired pad character:_

Enter the number of pad characters to send (0-255):_
```

Mit diesen beiden Fragen werden Sie aufgefordert, das Zeichen zu bestimmen, nach dem vom System *Füllzeichen* (Pad Character) gesendet werden sollen. Die Zahl der zu sendenden Zeichen können Sie mit der zweiten Antwort bestimmen.

Diese Möglichkeit ist nur für ältere Drucker gedacht, die nicht sehr schnell arbeiten. Ein langsamer Drucker benötigt sehr viel mehr Zeit, um nach einem Zeilenvorschub wieder zur ersten Spalte zu gelangen. Damit keine Zeichen verlorengehen, werden in dieser Zeit die Füllzeichen gesendet, deren Zahl Sie mit der zweiten Antwort bestimmen. Die nächste Frage ist aus dem gleichen Grunde auch hauptsächlich für ältere Geräte gedacht.

The timeout value is used to give slow devices time to respond to Input/Output requests. A small value is usually sufficient, but a number from 0 to 255 can be entered.

Enter timeout value for COM1:

Mit diesem Wert können Sie bestimmen, wie lange das System auf eine *Antwort* des Peripheriegerätes wartet, bevor es eine Fehlermeldung ausgibt und den Vorgang abbricht. Diese Möglichkeit ist nicht nur für Drucker gedacht, sondern auch für andere Geräte. Falls erforderlich, reicht hier ein kleiner Wert. Sie können aber eine Zahl von 0 bis 255 eingeben, je nachdem wie langsam das Gerät ist. Die von Zenith Data Systems gelieferten Geräte benötigen diese Wartezeit nicht. Haben Sie hier einen Wert eingegeben (z. B. 0), erscheint wieder das erste Untermenü, und Sie können die vorgenommenen Änderungen speichern.

Umleitung eines Parallelausganges

Soll ein serieller Drucker an Ihren Zenith PC angeschlossen werden, ist es unbedingt erforderlich, den Parallelausgang zum seriellen Ausgang *umzuleiten.* Sie müssen daher nicht nur das Protokoll des seriellen Ausganges (COM) festlegen, sondern diesen auch anschließend mit dem parallelen Ausgang (LPT) verbinden. Rufen Sie das Programm auf

CONFIGUR

und es erscheint wieder das Hauptmenü:

CONFIGUR

```
              CONFIGUR Version 3.01
Copyright (C) 1985 Zenith Data Systems Corporation

Use one of the following options to configure a device

A: Configure LPT device
B: Configure COM device
C: Change automatic partition assignment flag

Enter selection (A-D):_
```

Geben Sie jetzt ein **A** ein, um einen parallelen Ausgang zu definieren. Es erscheint das Untermenü:

```
Use one of the following options to select the type of configuration

A. Map parallel output to serial output
B. Configure parallel device

C. Exit

Enter selection (A-C):
```

Geben Sie jetzt wieder ein **A** ein für *Datenumleitung.* Beachten Sie, daß der serielle Drucker anschließend nur einwandfrei arbeitet, wenn er mit einem *Null-Modem-Kabel* angeschlossen wird. Dieses Kabel erhalten Sie von Ihrem Händler.

Es erscheint jetzt das folgende Menü, unabhängig von der Anzahl der Ausgänge, immer mit allen drei Möglichkeiten:

```
Select the parallel port to be mapped.

A. LPT1
B. LPT2
C. LPT3

D. Exit

Enter selction (A-D):_
```

Geben Sie hier wieder ein **A** ein für den ersten Ausgang, und es wird folgendes Menü gezeigt:

```
Use one of the following to select a mapping for LPT1

A. no mapping
B. Map to COM1
C. Map to COM2

Enter one of the mapping values:_
```

In diesem Menü können Sie den *seriellen* Ausgang bestimmen, an den die Daten gesendet werden sollen (COM1 oder COM2). Sie können auch mit der Möglichkeit A eine vorherige Umleitung wieder rückgängig machen. Nach einer Eingabe werden Sie wieder zum ersten Untermenü zurückgeführt:

```
Use one of the following options to select the type of configuration

A. Map parallel output to serial output
B. Configure parallel device

C. Exit

Enter selection (A-C):
```

Sie können jetzt weitere Ausgänge definieren, umleiten oder mit der Möglichkeit C zum Hauptmenü zurückgelangen:

CONFIGUR

```
                CONFIGUR Version 3.01
Copyright (C) 1985 Zenith Data Systems Corporation

Use one of the following options to configure a device

A: Configure LPT device
B: Configure COM device
C: Change automatic partition assignment flag

Use one of the following to modify an existing system

D: Exit programm
E: Make changes to disk
F: Make changes to memory
G: Make changes to both disk and memory

Enter selection (A-F):_
```

Bevor Sie die vorgenommenen Änderungen abspeichern, überprüfen Sie noch einmal Ihre Eingaben. Haben Sie den seriellen Ausgang (COM1 oder COM2) bereits definiert, an den der Drucker angeschlossen werden soll? Haben Sie den parallelen Ausgang zum entsprechenden seriellen Ausgang umgeleitet? Nur wenn Sie beide Schritte korrekt ausgeführt haben, arbeitet der Drucker einwandfrei.

Speichern Sie die vorgenommenen Änderungen dann mit einer der Möglichkeiten **E, F** oder **G.**

Automatische Bezeichnung für Winchesterteilbereiche

Haben Sie im Hauptmenü ein C eingegeben, erscheint jetzt folgendes Bild:

```
The system is currently set for automatic partition assignment.

Use one of the following to select assignment code.

A. Automatic partition assignment
B. Manual partition assignment

C. No change, exit to main menu

Enter selection (A-C):
```

Die erste Zeile zeigt Ihnen die momentane Art der Bezeichnung. Mit Eingabe von **A** oder **B** können Sie dies verändern. Bei Eingabe von **C** verlassen Sie diesen Teil des Programmes ohne Änderung.

Für die Programme Topview, MS-Net und Microsoft Project muß das Betriebssystem auf automatische Bezeichnung der Winchesterteilbereiche eingerichtet sein. Dies bestimmen Sie durch Eingabe von **A**. Diese Programme fragen nach ihrem Aufruf von MS-DOS die Zahl der vorhandenen Laufwerke ab und überprüfen diese sofort.

Das Betriebssystem hat bislang für jede Winchester vier Laufwerkbezeichnungen reserviert. Der erste Teilbereich wurde während des Systemstartes bereits bezeichnet, alle anderen können Sie mit dem Programm ASSIGN ebenfalls bezeichnen. Dies ist nicht erforderlich und geschieht nicht immer. Die Programme Topview, MS-Net und Microsoft Project können dann nicht arbeiten. Daher wurde dieser Menüpunkt neu geschaffen.

Haben Sie das Betriebssystem auf automatische Bezeichnung gestellt, wird immer nur ein Teilbereich des Winchesterlaufwerkes bezeichnet. MS-DOS reserviert keine weiteren Buchstaben für die anderen Bereiche. Mit dem Programm ASGNPART können Sie aber beliebig oft die Teilbereiche wechseln. Dabei gilt, daß immer nur auf einen Teilbereich der Winchester gleichzeitig zugegriffen werden kann.

Haben Sie mit der Eingabe von **B** das Betriebssystem auf manuelle Bezeichnung gestellt, können Sie mit Ihrer Winchester wie bisher arbeiten.

Während des Systemstartes wird nur der erste Teilbereich bezeichnet. Mit dem Programm ASGNPART können Sie die restlichen Bereiche bezeichnen und anschließend auf alle vier Bereiche gleichzeitig zugreifen.

Ändern Sie die bisherige Einstellung, müssen Sie im Hauptmenü anschließend mit den Buchstaben E oder G diese Änderungen speichern. Nach einem erneuten Systemstart wird dann die neue Art der Bezeichnung berücksichtigt.

Fehlermeldungen

Generell:

Nachdem Sie mit dem Programm CONFIGUR einen Ausgang für ein bestimmtes Gerät angepaßt haben, kann folgender Fehler auftauchen:

Das Peripheriegerät arbeitet nicht und Ihr Computer nimmt ebenfalls keine weiteren Befehle mehr an. Keine Sorge, es ist nichts defekt. Sie haben lediglich das Gerät nicht richtig angepaßt. Starten Sie das System neu (danach arbeitet alles wieder einwandfrei), und arbeiten Sie das Programm CONFIGUR noch einmal durch.

Cannot locate file IO.SYS Press any key to continue...	Die Datei IO.SYS ist nicht vorhanden. Betätigen Sie eine Taste.

ERKLÄRUNG:
Das Programm findet die Datei IO.SYS nicht. Nach Betätigen einer Taste sind Sie wieder im Betriebssystem.

IO.SYS file version incorrect Press any key to continue...	Die Version der Datei IO.SYS ist nicht kompatibel zum Programm.

ERKLÄRUNG:
Die vorhandene Datei IO.SYS kann vom Programm nicht modifiziert werden. Nach Betätigen einer Taste sind Sie wieder im Betriebssystem.

No serial or parallel devices are attached. No configuration is possible.	Es sind keine seriellen oder parallelen Ausgänge angeschlossen. Eine Anpassung ist nicht möglich.

ERKLÄRUNG:
Das Programm fragt erst die angeschlossenen Ausgänge ab. Es muß wenigstens ein serieller oder paralleler Ausgang vorhanden sein, damit das Programm abgearbeitet werden kann.

Version mismatch with IO.SYS	Falsche Version der Datei IO.SYS

ERKLÄRUNG:
Die vorhandene Version der Datei IO.SYS kann vom Programm nicht angepaßt werden.

COPY **Intern**

Zweck

kopiert eine oder mehrere Dateien

Syntax

COPY [/X] Dateibez [/X] [D:] Dateiname [/X]
COPY [/X] Dateibez D: [/X]
COPY [/X] Dateibez [D:] Weg [/X]

dabei bedeutet:

Dateibez Name der zu kopierenden Datei

D: Name des Ziellaufwerkes für die kopierte Datei

Dateiname neuer Name für die kopierte Datei

Weg mehrere Verzeichnisse in Folge in der Form:
[\] [Verzeichnis [\Verzeichnis...]]

/X Kombination aus den Zusatzeingaben:

- **/A** bearbeitet die folgende Datei als ASCII-Datei
- **/B** bearbeitet die folgende Datei als Binärdatei
- **/V** überprüft die kopierte Datei anschließend auf Fehler

Bemerkungen

Mit diesem Befehl können Sie Dateien und Daten *kopieren*. Die Dateien können dabei gleichzeitig umbenannt werden. Sie können die Dateien in dasselbe Verzeichnis oder in ein anderes Verzeichnis und/oder auf ein anderes Laufwerk kopieren.

Daten können mit diesem Befehl zwischen den Peripheriegeräten kopiert werden.

Ursprungsdatei

Die *Ursprungsdatei* ist die Datei mit den zu kopierenden Daten. Nach dem Befehlswort COPY muß immer der Name der Ursprungsdatei eingegeben werden. Dem Namen darf eine Laufwerkbezeichnung und/oder der Name eines Verzeichnisses vorangestellt werden. Der Name der Ursprungsdatei darf die Zeichen zur Zusammenfassung mehrerer Dateien („*" und „ ? ") enthalten. Wird dem Namen kein Laufwerk vorangestellt, wird die Datei auf dem Arbeitslaufwerk gesucht. Ebenso wird das momentane Verzeichnis abgesucht, stellen Sie dem Namen der Ursprungsdatei keinen Verzeichnisnamen voran. Bezeichnen Sie die Ursprungsdatei mit der Zeichenfolge „ *.* " und vorangestelltem Verzeichnis, bestimmen Sie alle Dateien dieses Verzeichnisses zum Kopieren.

Zieldatei

Sie haben verschiedene Möglichkeiten, die *Zieldatei* zu benennen. Sie können den alten Dateinamen übernehmen oder der Datei einen neuen Namen geben. Sie können dem Zieldateinamen folgende Zusätze voranstellen:

- Laufwerk
- Verzeichnis
- Laufwerk und Verzeichnis

Geben Sie für die Zieldatei eine Laufwerkbezeichnung ein, kopieren Sie die Datei mit dem gleichen Namen in das momentane Verzeichnis des genannten Laufwerkes.

Geben Sie für die Zieldatei einen Verzeichnisnamen ein, kopieren Sie die Datei mit dem gleichen Namen in das genannte Verzeichnis des momentanen Laufwerkes.

Geben Sie für die Zieldatei eine Laufwerkbezeichnung und einen Verzeichnisnamen ein, kopieren Sie die Datei auf das neue Laufwerk in das benannte Verzeichnis.

Bei allen drei Möglichkeiten können Sie der Zieldatei einen neuen Namen geben. Stellen Sie dem Dateinamen ein Verzeichnis voran, muß dieses Verzeichnis tatsächlich vorhanden sein, andernfalls wird die kopierte Datei mit diesem Verzeichnisnamen benannt.

Benennen Sie weder eine Zieldatei noch einen Zielort (Laufwerk oder Verzeichnis), kann der Befehl nicht ausgeführt werden. Es erscheint folgende Fehlermeldung:

File cannot be copied onto itself 0 File(s) copied	Datei kann nicht in sich selber kopiert werden Keine Datei kopiert

Wird die Datei korrekt kopiert, erscheint anschließend folgende Meldung:

n File(s) copied	n Dateien kopiert

n bezeichnet die Anzahl der kopierten Dateien. Nach dieser Meldung erscheint das Prompt des Betriebssystemes und der Befehl ist beendet.

Zusatzeingaben

Mit verschiedenen Zusatzeingaben können Sie die Arbeitsweise des Befehles COPY beeinflussen. Die Zusätze /A und /B müssen immer vor die Datei gestellt werden, auf die sie sich beziehen sollen. Der Zusatz /V wird immer nachgestellt.

/A - kopiere als ASCII-Datei

Geben Sie den Zusatz /A vor der Ursprungs- oder Zieldatei ein, auf den er sich beziehen soll. Der Befehl COPY kopiert dann die Ursprungsdatei bis zum ersten Zeichen ^Z, dem Zeichen für das Ende einer Textdatei. Wird der Zusatz vor einer Zieldatei eingegeben, fügt das Betriebssystem immer das Zeichen ^Z an. Kopieren Sie Daten zwischen Peripheriegeräten, führt MS-DOS dies immer als Kopie von ASCII-Dateien aus.

/B - kopiere als Binärdatei

Geben Sie den Zusatz /B vor einer Ursprungs- oder Zieldatei ein, auf den er sich beziehen soll. Der Befehl COPY kopiert dann die Ursprungsdatei immer mit der Zahl der Zeichen, die im Verzeichniseintrag gespeichert sind. Das Zeichen ^Z wird dabei nicht als Zeichen für das Dateiende erkannt. Geben Sie keinen Zusatz ein, kopiert MS-DOS Dateien auf andere Laufwerke immer als Binärdateien.

/V - überprüfe Datei

Geben Sie den Zusatz /V mit ein, vergleicht DOS laufend die gerade kopierten Daten der Zieldatei mit der Ursprungsdatei. Dadurch wird der Kopiervorgang erheblich verlangsamt. Entdeckt das Betriebssystem einen Fehler in der Zieldatei, wird der entsprechende Sektor noch einmal kopiert. Damit erreichen Sie, daß wichtige Daten garantiert fehlerfrei kopiert werden. Der Zusatz /V ruft die gleiche Funktion auf, wie der Befehl VERIFY, allerdings nur für einen Kopiervorgang.

Beispiele

Die folgenden Abschnitte zeigen verschiedene Beispiele für das Arbeiten mit dem Befehl COPY. Es werden Dateien sowohl auf ein anderes Laufwerk als auch in ein anderes Verzeichnis kopiert. In einem Beispiel wird die Datei umbenannt. Ebenso wird gezeigt, wie verschiedene Ursprungsdateien in eine Zieldatei kopiert werden können. Das letzte Beispiel kopiert Daten zu einem Peripheriegerät.

Beispiel 1

In diesem Beispiel wird die Datei auf ein anderes Laufwerk kopiert. Auf dem Arbeitslaufwerk (C:) ist im momentanen Verzeichnis die Datei WS.COM vorhanden. Die Datei soll auf das Laufwerk B: kopiert werden. Der Name wird nicht geändert. Geben Sie ein:

```
COPY WS.COM B:
```

Es flackern jetzt kurz die beiden Kontrollampen der Laufwerke auf, dann erscheint die Meldung:

1 File(s) copied	1 Datei kopiert

Die Datei ist in das aktuelle Verzeichnis des Laufwerkes B: kopiert worden. Sie können dies mit dem Befehl DIR kontrollieren.

Beispiel 2

In diesem Beispiel wird die Datei WS.COM in ein anderes Verzeichnis des Arbeitslaufwerkes kopiert. Die Datei wird nicht umbenannt. Geben Sie ein:

```
COPY WS.COM TEXT
```

Das Subverzeichnis TEXT ist vom momentanen Verzeichnis aus angelegt worden. In dieses Verzeichnis ist die Datei WS.COM kopiert worden. Es erscheint ebenfalls wieder die Meldung:

1 File(s) copied	1 Datei kopiert

Kopieren Sie Dateien in ein anderes Verzeichnis, wird das Verzeichnis wie ein Laufwerk adressiert. Achten Sie darauf, daß das Verzeichnis tatsächlich vorhanden ist. Findet MS-DOS dieses Verzeichnis nicht, benennt es automatisch die neue Datei um. Die Datei erhält dann den Namen des angeblichen Verzeichnisses.

Beispiel 3

In diesem Beispiel wird die zu kopierende Datei gleichzeitig umbenannt. Geben Sie ein:

```
COPY DATEI.ALT D:\BASIC\DATEI.NEU
```

Die Datei DATEI.ALT des momentanen Verzeichnisses wird auf das Laufwerk D: in das Verzeichnis \BASIC (erstes Verzeichnis nach dem Rootdir) kopiert und erhält den Namen DATEI.NEU.

Mit dieser Möglichkeit können Sie eine Datei auch in dasselbe Verzeichnis kopieren. Geben Sie ein:

```
COPY DATEI.ALT DATEI.NEU
```

Sie haben dann im Arbeitsverzeichnis zwei Dateien mit gleichem Inhalt, aber verschiedenen Namen.

Beispiel 4

Im folgenden Beispiel werden mehrere Ursprungsdateien in eine Zieldatei kopiert. Dies ist sinnvollerweise nur mit Textdateien möglich. Sollen z. B. alle Dateien mit dem Zusatz .TXT zusammenkopiert werden, geben Sie ein:

```
COPY *.TXT TEXT.DOC
```

Es werden jetzt sämtliche Dateien des momentanen Verzeichnisses mit dem Zusatz .TXT in die Datei TEXT.DOC kopiert. Während des Kopiervorganges werden die einzelnen Dateinamen auf dem Bildschirm angezeigt. Zum Schluß erscheint die Meldung:

```
1 File(s) copied                1 Datei kopiert
```

Damit wird angezeigt, daß eine neue Datei erzeugt wurde. Achten Sie bei dieser Art der Befehlseingabe darauf, daß der Name der Zieldatei nicht durch die globale Bezeichnung der Ursprungsdateien mit erfaßt wird. Im obigen Beispiel wurde daher die neue Datei mit dem Namenszusatz .DOC versehen.

Wäre die neue Datei ebenfalls mit dem Zusatz .TXT bezeichnet worden, wäre folgende Fehlermeldung erschienen:

Content of destination lost before copy	Inhalt der Kette vor dem Kopieren verloren.

Kopieren Sie mehrere Dateien zusammen, vergleicht das System jeden Namen der Ursprungsdatei mit dem Namen der Zieldatei. Bei Namensgleichheit müßte die Datei in sich selber kopiert werden. Da dies ist nicht möglich ist, wird die Datei aus der Liste der Ursprungsdateien gestrichen und obige Meldung ausgegeben. Es wird dann die nächste Datei kopiert.

Kopieren Sie die Dateien mit folgenden Befehl zusammen, erscheint die Fehlermeldung nicht:

```
COPY DOC.TXT+*.TXT
```

Da Sie in diesem Beispiel keine Zieldatei benannt haben, werden alle zu kopierenden Dateien mit dem Zusatz .TXT an die erste Ursprungsdatei (in diesem Falle DOC.TXT) angehängt. Existiert die Datei DOC.TXT noch nicht, wird sie angelegt.

Beispiel 5

Mit dem Befehl COPY können Sie auch Daten zwischen den Peripheriegeräten kopieren. Sie können z. B. eine Datei auf den Bildschirm kopieren. Geben Sie ein:

```
COPY DOC.TXT CON
```

Jetzt wird die Datei DOC.TXT von der Diskette oder Winchester auf den Bildschirm kopiert. Dies ist im Prinzip die Funktion des Befehls TYPE. Ein weiteres Beispiel:

```
COPY DOC.TXT PRN
```

Dieser Befehl kopiert die Datei DOC.TXT zum Drucker. Dies ist im Prinzip die Funktion des Befehls PRINT. Ein weiteres Beispiel:

COPY

COPY CON AUTOEXEC.BAT

Mit diesem Befehl kopieren Sie alle Daten, die Sie über die Tastatur (CON) eingeben, in die Datei AUTOEXEC.BAT. Beenden Sie die Eingabe in die Datei mit dem Zeichen CTRL-Z und **RETURN.** Es erscheint dann ebenfalls die Meldung:

1 File(s) copied	1 Datei kopiert

Fehlermeldungen

Content of destination lost before copy	Inhalt der Kette vor dem Kopieren verloren.

ERKLÄRUNG:
Dieser Fehler tritt auf, wenn mit einem Kopierbefehl mehrere Dateien zusammenkopiert werden. Benennen Sie die neue Datei dabei mit einem Namen der zu kopierenden Dateien, so tritt dieser Fehler während des Kopiervorganges auf. Die betreffende Datei wird nicht mitkopiert. Die restlichen Dateien werden korrekt kopiert.

File cannot be copied onto itself 0 File(s) copied	Datei kann nicht auf sich selber kopiert werden. Keine Datei kopiert.

ERKLÄRUNG:
Die neue Datei hat den gleichen Namen wie die alte. Damit würde die Datei auf sich selber kopiert werden. Das ist nicht möglich. Geben Sie einen anderen Dateinamen ein, oder kopieren Sie die Datei in ein anderes Verzeichnis.

File creation error	Falsche Dateianlage

ERKLÄRUNG:
Im Inhaltsverzeichnis der Diskette oder des Winchesterlaufwerkteiles ist kein Platz mehr, um eine weitere Datei zu speichern. Sofern noch freier Speicherplatz auf der Platte vorhanden ist, müssen im Verzeichnis Einträge gelöscht werden, damit eine weitere Datei gespeichert werden kann.

... File not found 0 File(s) copied	Dateiname nicht gefunden keine Datei kopiert

ERKLÄRUNG:
Der eingegebene Dateiname ist im Verzeichnis nicht vorhanden. Überprüfen Sie Ihre Eingabe.

Insufficient disk space 0 File(s) copied	Nicht genug Speicherkapazität auf der Platte

ERKLÄRUNG:
Die Speicherkapazität der Platte, auf die geschrieben werden soll, reicht nicht aus für die Datei. Wechseln Sie die Diskette (Teilbereich der Winchester).

Invalid directory	Falsches Inhaltsverzeichnis

ERKLÄRUNG:
Mindestens eins der Verzeichnisse des Weges ist nicht korrekt eingegeben worden.

Invalid drive name	Falsche Laufwerkbezeichnung

ERKLÄRUNG:
Das Ursprungs- oder Ziellaufwerk ist nicht richtig benannt worden, oder es ist nicht vorhanden.

Invalid number of parameters	Falsche Anzahl von Zusatzeingaben

ERKLÄRUNG:
Sie haben zu viele oder zu wenige Zusatzeingaben gemacht. Überprüfen Sie Ihre Eingabe.

Not ready error reading drive D: Abort, Retry, Ignore?	Lesefehler im Laufwerk D: Abbrechen,Wiederholen, Ignorieren

ERKLÄRUNG:
Das Ziellaufwerk enthält keine Diskette oder die Klappe des Laufwerkes ist offen. Beheben Sie den Fehler, und wiederholen Sie den Vorgang mit Eingabe von **R.** Ist das Ziellaufwerk nicht vorhanden, brechen Sie den Vorgang durch Eingabe von **A** ab. Geben Sie dann den Befehl neu ein.

CTTY **Intern**

Zweck

ändert das Peripheriegerät, von dem aus Befehle eingegeben werden.

Syntax

CTTY Device

dabei bedeutet:

Device ein Peripheriegerät angeschlossen an einem der folgenden Ein/Ausgänge:

AUX	Hilfseingang
CON	Tastatur für Eingabe, Bildschirm für Ausgabe
COM1	erster serieller Ausgang
COM2	zweiter serieller Ausgang (sofern vorhanden)

Bemerkungen

Mit diesem Befehl können Sie das Peripheriegerät ändern, von dem aus Sie Befehle eingeben. Sie können damit ein Modem oder ein anderes Terminal an Ihren Zenith PC anschließen. Alle Ein- und Ausgaben werden jetzt hiermit vorgenommen. Es dürfen nur Geräte angeschlossen werden, die sowohl Eingaben vornehmen können, als auch Ausgaben anzeigen können. Daher ist es z. B. nicht möglich, mit diesem Befehl einen Drucker zu benennen.

Sie schalten auf ein anderes Gerät um, indem Sie z. B. eingeben:

```
CTTY AUX
```

CTTY

Jetzt werden alle Ein-/Ausgaben von dem Gerät vorgenommen, das am Ausgang AUX Ihres PC angeschlossen ist, dies kann z. B. ein anderes Terminal sein. Von diesem Gerät aus muß auch anschließend wieder der Befehl zum Zurückschalten kommen. Geben Sie ein:

```
CTTY CON
```

Weitere Erläuterungen zu möglichen Peripheriegeräten finden Sie im Kapitel 1.

Mit dem Befehl CTTY können Sie jeden Charaktertreiber ansprechen. Sie haben damit die Möglichkeit, Ihren eigenen Treiber zu schreiben und mit einem eigenen Namen zu versehen.

DATE Intern

Zweck

zeigt und ändert das Tagesdatum

Syntax

```
DATE
DATE mm-tt-jj
DATE tt-mm-jj
```

dabei bedeutet:

mm	Eingabe von 1 bis 12 für den Monat
tt	Eingabe von 1 bis 31 für den Tag
jj	Eingabe von 80 bis 99 oder 1980 bis 2099 für das Jahr

Bemerkungen

Das Datum wird vom System bei allen Eingabefunktionen für Dateien benutzt. Damit wird Ihnen immer angezeigt, wann zuletzt in die Datei geschrieben wurde.

Das Datumsformat können Sie mit dem Befehl COUNTRY in der Datei CONFIG.SYS bestimmen. Haben Sie mit dem Befehl SELECT Ihre Systemdisketten kopiert, wurde automatisch die Datei mit einem entsprechenden Befehl angelegt. Weitere Informationen dazu finden Sie im Kapitel 3 oder 9.

Der Befehl kann sowohl direkt eingegeben, als auch über eine Batchdatei aufgerufen werden. Rufen Sie die Funktion durch einfache Befehlseingabe auf, erscheint die Meldung:

Current date is day mm-dd-yy	Derzeitiges Datum ist Tag MM-TT-JJ
Enter new date:	Neues Datum:

Geben Sie dann das Datum in der amerikanischen Schreibweise ein:

Monat	erlaubt sind Zahlen zwischen 1 und 12
Tag	erlaubt sind Zahlen zwischen 1 und 31
Jahr	erlaubt sind Zahlen zwischen 80 und 99 für die Jahre 1980 bis 1999 und vierstellige Zahlen für die Jahre 1980 bis 2099

Die Zahlen dürfen nur durch einen Schrägstrich (/) oder durch einen Bindestrich (-) getrennt werden. Bei einer falschen Eingabe erscheint die Fehlermeldung:

Invalid date	Falsches Datum
Enter new date:	Neues Datum:

Soll das bestehende Datum nicht geändert werden, übernehmen Sie es durch einfaches Betätigen der Taste **RETURN**. Der Wochentag wird durch das System automatisch errechnet, er kann von Ihnen nicht eingegeben werden.

Sie haben auch die Möglichkeit, das Datum direkt in der Befehlszeile miteinzugeben. Ein Beispiel:

```
DATE 26-11-85
```

ist eine direkte Eingabe für den 26. November 1985. Das System gibt dann keine weitere Meldung aus.

DEL oder ERASE Intern

Zweck

löscht Dateien

Syntax

DEL Dateibez
DEL [D:] Weg [\Dateiname]
ERASE Dateibez
ERASE [D:] Weg [\Dateiname]

dabei bedeutet:

Dateibez Name der zu löschenden Datei mit Laufwerk und/ oder Verzeichnisname

D: Name des Laufwerkes mit der zu löschenden Datei

Weg mehrere Verzeichnisse in Folge in der Form:

[\] Verzeichnis [\Verzeichnis...]]

Dateiname Name der zu löschenden Datei

Bemerkungen

Mit diesem Befehl löschen Sie mehrere Dateien. Sie können den Befehl mit beiden Namen, DEL oder ERASE, eingeben. Die Funktionen sind gleich.

Der Befehl DEL löscht die Dateien nicht tatsächlich auf der Platte. Es wird nur das erste Zeichen des Dateinamens mit einem besonderen Wert überschrieben. Daran erkennt das System, daß dieser Eintrag gelöscht sein soll. Erst der nächste Neueintrag überschreibt die Daten wirklich (s. a. FORMAT).

Damit der Befehl ausgeführt werden kann, müssen Sie nach dem Befehlsnamen DEL oder ERASE mindestens den Namen der zu löschenden Datei eingeben. Geben Sie den Befehl ohne weitere Zusätze ein, erscheint die Fehlermeldung:

Invalid number of parameters	Falsche Anzahl der Zusatzeingaben

Geben Sie den Namen einer Datei ein, sucht das System die Datei im momentanen Verzeichnis. Ist die Datei vorhanden und kann gelöscht werden, erscheint nach der Befehlsausführung ohne weitere Meldung das Prompt des Betriebssystemes. Damit ist die Datei gelöscht.

Befindet sich die Datei nicht im momentanen Verzeichnis oder Laufwerk, stellen Sie dem Dateinamen die entsprechenden Bezeichnungen voran. Das System sucht dann die Datei im angegebenen Verzeichnis und löscht sie dort. Ist der letzte Name der Befehlszeile ein Verzeichnisname, werden alle Dateien dieses Verzeichnisses gelöscht.

Beispiele

Ist im momentanen Verzeichnis die Datei BRIEF.PRV gespeichert und soll gelöscht werden, geben Sie ein:

```
DEL BRIEF.PRV
```

Nach einigen Sekunden erscheint wieder das Prompt und zeigt Ihnen somit an, daß die Datei gelöscht ist. Überprüfen Sie dies mit dem Befehl DIR.

Ist die Datei im Verzeichnis \TEXT\SPEICHER vorhanden und soll gelöscht werden, geben Sie ein:

```
DEL \TEXT\SPEICHER\BRIEF.PRV
```

Auch jetzt wird durch das Prompt des Betriebssystemes angezeigt, daß die Datei gelöscht wurde. Eine besondere Meldung erscheint nicht.

DEL oder ERASE

Befindet sich die zu löschende Datei nicht auf dem momentanen Laufwerk, muß die Laufwerkbezeichnung dem Dateinamen vorangestellt werden. Ein Beispiel:

```
DEL D:BRIEF.PRV
```

Im aktuellen Verzeichnis des Laufwerkes D: wird jetzt die Datei BRIEF.PRV gelöscht. Ist die Datei nicht vorhanden, erscheint folgende Fehlermeldung:

```
File not found                  Datei nicht vorhanden
```

Sie können mit dem Befehl DEL auch mehrere Dateien gleichzeitig löschen. Fassen Sie die Dateien dann mit dem Globalzeichen zusammen. Ein Beispiel:

```
DEL A:*.DOC
```

Jetzt werden sämtliche Dateien des Laufwerkes A:, deren Zusatz .DOC ist, gelöscht.

ACHTUNG: Überzeugen Sie sich vorher, daß alle so zusammengefaßten Dateien auch wirklich gelöscht werden dürfen. Die Dateien sind sonst sehr schnell unwiderruflich verloren.

Benutzen Sie die Zeichenfolge „ *.* ", um alle Dateinamen zu erfassen, erscheint vor der Befehlsausführung die Frage:

```
Are you sure (Y/N)?             Sind Sie sicher (J/N)?
```

Nur nach Eingabe von **Y** für Ja und **RETURN** wird der Befehl ausgeführt. Alle anderen Eingaben brechen den Befehl ab. Dies gilt ebenfalls, wenn Sie ein Verzeichnis zum Löschen benennen. Ein Beispiel:

```
DEL C:\TEXT
```

Mit diesem Befehl löschen Sie alle Dateien des Verzeichnisses TEXT im Laufwerk C:. Zur Sicherheit erscheint auch hier noch einmal die Frage: Are you sure?

Fehlermeldungen

File not found	Datei nicht vorhanden

ERKLÄRUNG:
Sie haben einen Dateinamen zum Löschen benannt, der nicht vorhanden ist oder der nicht im benannten Verzeichnis vorhanden ist. Ebenso ist es möglich, daß Sie den Namen falsch geschrieben haben. Überprüfen Sie Ihre Eingabe.

Invalid directory	Falsches Inhaltsverzeichnis

ERKLÄRUNG:
Mindestens eins der Verzeichnisse des Weges ist nicht korrekt eingegeben worden.

Invalid number of parameters	Falsche Anzahl der Zusatzeingaben

ERKLÄRUNG:
Sie haben zu viele oder zu wenige Zusatzeingaben gemacht. Überprüfen Sie Ihre Eingabe.

Not ready error reading drive D: Abort, Retry, Ignore?	Lesefehler im Laufwerk D: Abbrechen, Wiederholen, Ignorieren

ERKLÄRUNG:
Das Ziellaufwerk enthält keine Diskette oder die Klappe des Laufwerkes ist offen. Beheben Sie den Fehler und wiederholen Sie den Vorgang mit Eingabe von **R**. Ist das Ziellaufwerk nicht vorhanden, brechen Sie den Vorgang durch Eingabe von **A** ab. Geben Sie dann den Befehl neu ein.

DIR Intern

Zweck

zeigt Dateien des Inhaltsverzeichnisses mit den gespeicherten Informationen auf dem Bildschirm an.

Syntax

DIR [D:] [Dateiname] [/X]
DIR [D:] [Weg] [/X]

dabei bedeutet:

D: Name des Laufwerkes, dessen Inhaltsverzeichnis angezeigt werden soll

Dateiname Name(n) der Datei(en), deren Informationen gezeigt werden sollen

Weg mehrere Verzeichnisse in Folge in der Form:

[\] [Verzeichnis [Verzeichnis...]]

Bemerkungen

Mit diesem Befehl können Sie sich Dateieinträge des Inhaltsverzeichnisses anzeigen lassen. Es werden folgende Informationen zu jeder Datei gezeigt:

- Name
- Namenszusatz
- Größe in Bytes
- Datum
- Uhrzeit

Datum und Uhrzeit zeigen Ihnen immer an, wann die Datei erstellt oder zuletzt geändert wurde.

Geben Sie den Befehl ohne weitere Zusätze ein, wird das komplette Verzeichnis des momentanen Laufwerkes angezeigt.

Geben Sie ein

DIR

und es könnte folgendes Verzeichnis erscheinen:

```
Volume in drive C is ZENITH 1
Directory of C: \

KAP11B                    56704  8-16-86   5:44p
KAPH                      24192  8-08-86  10:36p
KAP11C2                   23808  8-09-86   7:06p
BEFEHLE       <DIR>              8-02-86   9:15a
DEL           FEH         1280   8-28-86   5:52a
COPY          FEH         3072   8-28-86   5:46a
DISKCOMP      FEH         1408   8-28-86   6:13a
DISKCOPY      FEH         1792   8-28-86   7:18a
DIR           FEH         384    8-28-86   5:37p
              9 File(s)   245760 bytes free
```

Außer den Dateiinformationen erhalten Sie auch Angaben zum Laufwerk. Es wird der Name des Laufwerkes (ZENITH 1), sowie der Name des momentanen Verzeichnisses (\) mit angezeigt. In der letzten Zeile sehen Sie die Zahl der gezeigten Dateien (9) und die noch freie Speicherkapazität des Laufwerkes (245760 Bytes).

Geben Sie nach dem Befehl den Namen einer Datei ein, wird nur noch diese bestimmte Datei angezeigt. Verwenden Sie im Dateinamen die Zeichen zur Zusammenfassung mehrerer Dateien, wird die so bezeichnete Dateigruppe angezeigt. Geben Sie z. B. ein

DIR *.FEH

werden vom oberen Beispiel nur noch folgende Dateien gezeigt:

```
Volume in drive C is ZENITH 1
Directory of C: \

DEL           FEH         1280   8-28-86   5:52a
COPY          FEH         3072   8-28-86   5:46a
DISKCOMP      FEH         1408   8-28-86   6:13a
DISKCOPY      FEH         1792   8-28-86   7:18a
DIR           FEH         384    8-28-86   5:37p
              5 File(s)   245760 bytes free
```

Die Informationen über das Laufwerk werden immer komplett angezeigt.

Geben Sie an Stelle des Dateinamens den Namen eines Verzeichnisses ein, werden alle Einträge dieses Verzeichnisses gezeigt.

Werden mehr als 23 Einträge gezeigt, rollen die ersten Einträge bereits aus dem Bild. Sie können dann die Ausgabe mit der Tastenkombination **CTRL-S** anhalten oder den Befehl mit dem Filter MORE kombinieren. Sie können aber auch einen oder beide Befehlszusätze eingeben, um die Bildschirmausgabe zu beeinflussen.

/P - Ausgabe seitenweise

Der Zusatz /P unterbricht die Ausgabe nach 23 Dateien. In der letzten Zeile erscheint dann die Aufforderung:

```
Strike any key when ready          Betätige Taste, wenn fertig
```

Haben Sie alle Einträge gesehen, betätigen Sie eine beliebige alphanumerische Taste. Die Ausgabe wird fortgesetzt und nach maximal 23 weiteren Einträgen erneut angehalten.

/W - Ausgabe weit

Der Zusatz /W zeigt die Dateien in leicht veränderter Form. Es werden jetzt immer 5 Dateien in einer Zeile angezeigt. Dabei werden nur noch der Name und der Zusatz gezeigt, alle anderen Angaben entfallen. Der Zusatz /W kann gleichzeitig mit dem Zusatz /P eingegeben werden.

Geben Sie z. B. folgenden Befehl ein:

```
DIR /W
```

Es könnte folgendes Bild erscheinen:

```
Volume in drive C is HANDBUCH
Directory of C:\system

.                ..                IO        SYS   MSDOS     SYS   COMMAND   COM
FORMAT   COM     CONFIGUR  COM     CHKDSK    COM   EDLIN     COM   PRINT     COM
SYS      COM     DISKCOPY  COM     DISKCOMP  COM   DEBUG     COM   FC        EXE
RECOVER  COM     RDCPM     COM     SEARCH    COM   APPLY     COM   CIPHER    COM
SORT     EXE     MORE      COM     FIND      EXE   ASSIGN    COM   PSCIDS    COM
PSCOKI   COM     PSCTS315  COM     PSCP920   COM   PSCMX80   COM   PSCMPI    COM
ANSI     SYS     AUTOEXEC  BAT     PART      EXE   PREP      EXE   DETECT    COM
BACKUP   EXE     RESTORE   EXE     SHIP      COM   LINK      EXE   LIB       EXE
EXE2BIN  EXE     BSPDGR    COM
         42 File(s)    3932160 bytes free
```

Sie können mit dem Befehl DIR jedes beliebige Verzeichnis der Platte anzeigen lassen, unabhängig vom momentanen Verzeichnis. Geben Sie nach dem Befehl den Namen des gewünschten Verzeichnisses ein, werden alle gespeicherten Dateien angezeigt. Sie können als Verzeichnisnamen auch die Kurzbezeichnung für das momentane Verzeichnis (.) oder das vorherige Verzeichnis (..) eingeben. Nähere Informationen dazu im Kapitel 7. Sind Sie in einem SUBDIR, geben Sie ein

```
DIR ..
```

und es werden sämtliche Einträge des vorherigen Verzeichnisses angezeigt.

Sowie Sie den Befehl mit dem Filter SORT verketten können (s. a. Seite 11.221), ist es möglich, die Ausgabe auf den Drucker umzuleiten. Ein Beispiel:

```
DIR > PRN
```

gibt die Informationen nicht mehr auf den Bildschirm, sondern auf den Drucker. Der Ausdruck kann dann z. B. auf die Diskettenhülle geklebt werden.

Fehlermeldungen

File not found — Datei nicht vorhanden

ERKLÄRUNG:
Die eingegebene Datei wurde nicht gefunden, sie ist eventuell nicht auf der Platte. Überprüfen Sie Ihre Eingabe.

Invalid directory — Falsches Inhaltsverzeichnis

ERKLÄRUNG:
Das angegebene Verzeichnis ist nicht vorhanden. Überprüfen Sie Ihre Eingabe.

DISKCOMP Extern

Zweck

überprüft zwei Disketten auf absolute Gleichheit

Syntax

[d:] [Weg] DISKCOMP [S: [D:]]

dabei bedeutet:

d:	vor dem Programmnamen bestimmt das aufzurufende Laufwerk
Weg	vor dem Programmnamen bestimmt das aufzurufende Verzeichnis
S:	Laufwerkname für die erste Diskette
D:	Laufwerkname für die zweite Diskette

Bemerkungen

Das Programm DISKCOMP kann im *Netzwerk* **nicht** benutzt werden.

Dieses Diensprogramm vergleicht die Daten von 2 Disketten, um Übereinstimmung festzustellen. Das Programm kann nur Disketten mit demselben Format vergleichen. Die Anzahl der Spuren und die Zahl der Sektoren je Spur müssen übereinstimmen, damit das Programm korrekt abläuft.

Haben Sie eine Diskette mit dem Befehl DISKCOPY kopiert, können Sie anschließend mit dem Programm DISKCOMP die neue Diskette überprüfen. Das Programm führt dann dieselbe Funktion aus, wie der Zusatz /V des Programmes DISKCOPY.

ACHTUNG:
Benutzen Sie dieses Programm nicht für Laufwerke, denen Sie mit dem Befehl ASSIGN vorübergehend einen anderen Namen zugewiesen haben. Durch die Umbenennung wird nicht auf das ursprüngliche Laufwerk zugegriffen, sondern auf das umgeleitete. Dadurch können die wahren Eigenschaften des Laufwerkes verdeckt werden, und das Programm arbeitet nicht fehlerfrei.

Programmaufruf

Sie haben zwei Möglichkeiten, das Programm aufzurufen:

- Sie geben alle Zusätze in der Befehlszeile ein.
- Sie geben die Zusätze auf eine entsprechende Anfrage ein.

Rufen Sie das Programm auf durch Eingabe von:

DISKCOMP

und es erscheint folgendes auf dem Bildschirm:

```
                    DISKCOMP Version 3.06
Copyright (C) 1985, Zenith Dyta Systems Corporation

Source1 drive name? (A-F)_:
```

Das Programm meldet sich mit seinem Namen, und fragt Sie nach der ersten Laufwerkbezeichnung. Es sind vorher alle vorhandenen Laufwerke abfgefragt worden, damit die Anfrage entsprechend gehalten werden kann. In diesem Falle sind 6 logische Laufwerke vorhanden (2 Diskettenlaufwerke und 4 Winchesterteilbereiche). Rufen Sie das Programm auf, könnten andere Laufwerknamen vorgegeben sein.

Geben Sie jetzt den Buchstaben des Laufwerkes mit der ersten Diskette oder den entsprechenden Winchesterteilbereich ein. Es erscheint danach die zweite Frage:

```
Source2 drive name? (A-F)_:
```

Geben Sie hier den entsprechenden Buchstaben für das zweite Laufwerk ein. Für beide Laufwerkbezeichnungen wird nur die Eingabe des Buchstabens gefordert. Der nachgestellte Doppelpunkt wird vom Programm automatisch eingesetzt.

Jetzt erscheint folgende Aufforderung:

```
Place the source1 disk in . and the source2 disk in ..
Press RETURN when ready.
```

Legen Sie die zu vergleichenden Disketten in die entsprechenden Laufwerke, und bestätigen Sie durch Betätigen der Taste **RETURN**.

Das Programm vergleicht beide Disketten und gibt dabei die Meldung aus:

```
Verifying...
```

Ist der Vergleich beendet, erscheint bei völliger Gleichheit der Disketten keine weitere Meldung. Wurden Untersdchiede festgestellt, wird nur die Meldung ausgegeben:

Disk verify failure	Diskettenprüffehler

Eine genauere Bezeichnung des Unterschiedes erfolgt nicht. Das Programm wird beendet mit der Frage:

Do you wish compare more disks (Y/N)? <N>	Möchten Sie weitere Disketten vergleichen (J/N)? <N>

Geben Sie ein **Y** für Ja ein, beginnt das Programm erneut, betätigen Sie nur die Taste **RETURN**, übernehmen Sie den vorgegebenen Wert **N**(ein) und verlassen das Programm. Es erscheint wieder das Prompt des Betriebssystemes.

Rufen Sie das Programm zusammen mit einem Laufwerknamen auf, wird dieses Laufwerk vom Programm als erste Laufwerkbezeichnung angenommen. Sie werden dann nur noch aufgefordert, den zweiten Laufwerknamen einzugeben.

DISKCOMP

Nach Eingabe von:

DISKCOMP A:

erscheint folgende Meldung:

```
                DISKCOMP Verson 3.06
Copyright (C) 1985, Zenith Data Systems Corporation

Source2 drive name? (A-F):_
```

Nach Eingabe einer weiteren Laufwerkbezeichnung erscheint die Aufforderung:

```
Place the source1 disk in A and the source2 disk ..
Press RETURN when ready.
```

Nach Betätigen der Taste **RETURN** beginnt der Vergleich.

Rufen Sie das Programm mit zwei Laufwerkbezeichnungen auf, erscheint sofort die Aufforderung, beide Disketten einzulegen. Geben Sie ein:

DISKCOMP A: B:

Es erscheint folgende Meldung:

```
                DISKCOMP Version 3.06
Copyright (C) 1985, Zenith Data Systems Corporation

Place the source1 disc in A and the source2 disk in B.
Press RETURN when ready.
```

Auch jetzt beginnt der Vergleich der Disketten nach Betätigen der Taste **RETURN**.

Weitere Bemerkungen

Mit diesem Programm ist es nur sinnvoll, identische Disketten zu vergleichen. Damit können Sie feststellen, ob mit dem Befehl DISKCOPY die Diskette richtig dupliziert wurde. Ebenso läßt sich prüfen, ob zu einem späteren Zeitpunkt die Diskette immer noch mit dem Original übereinstimmt.

Wollen Sie zwei Dateien auf Gleichheit überprüfen, benutzen Sie dafür das Dienstprogramm FC.

Fehlermeldungen

Disk verify failure	Diskettenprüffehler

ERKLÄRUNG:
Die beiden zu überprüfenden Disketten sind nicht identisch.

Drives specified must not be the same	Laufwerkangaben dürfen nicht gleich sein

ERKLÄRUNG:
Sie haben für beide Disketten das gleiche Laufwerk benannt. Dies ist nicht möglich. Das Programm wiederholt die Laufwerkanfragen:

Source1 drive name? (A-X):_	Erstes Laufwerk? (A-X):_

Achten Sie bei der Eingabe darauf, unterschiedliche Laufwerke zu bezeichnen.

Incompatible media. Cannot continue	Disketten nicht kompatibel Das Programm wird abgebrochen

ERKLÄRUNG:
Sie vergleichen zwei Disketten unterchiedlichen Typs. Beide Disketten müssen gleich formatiert sein (z. B. DS/DD oder mit 8 bzw. 9 Sektoren). Obwohl die Dateien auf beiden Disketten gleich sein könnten, ist es nicht möglich, den Vergleich durchzuführen.

DISKCOMP

Read error on Source1 drive	Lesefehler auf der ersten Diskette

ERKLÄRUNG:
Die Diskette 1 hat fehlerhafte Sektoren oder ist nicht von DOS formatiert worden.

Read error on Source2 drive	Lesefehler auf der zweiten Diskette

ERKLÄRUNG:
Die Diskette 2 hat fehlerhafte Sektoren oder ist nicht von DOS formatiert worden.

Cannot DISKCOMP a Network drive	Ein Laufwerk des Netzwerkes kann nicht verglichen werden

ERKLÄRUNG:
Sie haben versucht, mit dem Befehl DISKCOMP ein Laufwerk des Netzwerkes zu vergleichen. Dies ist nicht möglich.

DISKCOPY Extern

Zweck

erstellt ein genaues Duplikat der Ursprungsdiskette

Syntax

[d:] [Weg] DISKCOPY [S: [D:]] /V]

dabei bedeutet:

d: vor dem Programmnamen bestimmt das aufzurufende Laufwerk

Weg vor dem Programmnamen bestimmt das aufzurufende Verzeichnis

S: Laufwerk mit der Ursprungsdiskette

D: Laufwerk mit der Zieldiskette

/V: überprüft die Dateien der Zieldiskette auf Fehler

Bemerkungen

Das Programm DISKCOPY kann im *Netzwerk* **nicht** benutzt werden.

Mit diesem Dienstprogramm können Sie schnell und einfach Disketten duplizieren. Die neue Zieldiskette wird wie die Ursprungsdiskette formatiert und speichert die Dateien anschließend in der gleichen Reihenfolge. Benutzen Sie dieses Programm, um von Ihren Masterdisketten eine erste Kopie zu erstellen. Von dieser Kopie können Sie dann Ihre Arbeitskopie erstellen. Verfahren Sie so mit sämtlichen Anwenderprogrammen, können Sie immer wieder zerstörte Programme mit wenig Aufwand kopieren.

ACHTUNG:
Benutzen Sie dieses Programm nicht für Laufwerke, denen Sie mit dem Befehl ASSIGN vorübergehend einen anderen Namen zuge-

wiesen haben. Durch die Umbenennung wird nicht auf das ursprüngliche Laufwerk zurückgegriffen, sondern auf das umgeleitete. Dadurch können die wahren Eigenschaften des Laufwerks verdeckt werden, und das Programm arbeitet nicht fehlerfrei.

Programmaufruf

Sie haben zwei Möglichkeiten, das Programm aufzurufen:

- Sie geben alle Zusätze in der Befehlszeile ein.
- Sie geben die Zusätze auf eine entsprechende Anfrage ein.

Rufen Sie das Programm auf durch Eingabe von

```
DISKCOPY
```

und es erscheint folgendes auf dem Bildschirm:

```
                    DISKCOPY Version 3.06
Copyright (C) 1985, Zenith Data Systems Corporation

Source drive name? (A-F):_
```

Das Programm meldet sich mit seinem Namen und fragt Sie nach dem Laufwerk mit der zu kopierenden Diskette. Es sind vorher alle vorhandenen Laufwerke abgefragt worden, damit die Anfrage entsprechend gehalten werden kann. In diesem Falle sind 6 logische Laufwerke vorhanden (2 Diskettenlaufwerke und 4 Winchesterteilbereiche). Rufen Sie das Programm auf, könnten andere Laufwerknamen vorgegeben sein.

Geben Sie jetzt den Buchstaben des Laufwerkes mit der Ursprungsdiskette ein. Achten Sie unbedingt darauf, daß diese Diskette mit einem Schreibschutz versehen ist. Nur so können Sie verhindern, daß Ihre Daten irrtümlich gelöscht werden. Es erscheint danach die zweite Frage:

```
Destination drive name? (A-F):_
```

Geben Sie hier den entsprechenden Buchstaben für das Laufwerk mit der neuen Diskette ein. Dies sollte eine noch nicht benutzte Diskette sein, da sie anschließend formatiert wird. Eventuell vorhandene Daten sind dann gelöscht. Für beide Laufwerkbezeichnungen wird nun die Eingabe des Buchstabens gefordert. Der nachgestellte Doppelpunkt wird vom Programm automatisch eingesetzt.

Es erscheint jetzt die Aufforderung:

```
Place the source disc in . and the destination disk in ..
Press RETURN when ready
```

Legen Sie die Diskette mit den zu kopierenden Daten (Schreibschutz nicht vergessen) in das zuerst benannte Laufwerk und anschließend die neue Diskette in das zweite Laufwerk. Bestätigen Sie mit **RETURN**.

Das Programm formatiert zuerst die Zieldiskette. Die Diskette erhält das gleiche Format wie die Ursprungsdiskette. Es erscheint die Meldung:

Formatting destination...	Die Zieldiskette wird formatiert

Danach werden die Dateien kopiert. Dies wird durch die Meldung

Copying...	Kopierung...

angezeigt. Das Programm wird beendet mit der Frage:

Do you wish to copy another disk (Y/N)? <N>	Möchten Sie weitere Disketten kopieren (J/N)? <N>

Geben Sie ein **Y** für Ja ein, beginnt das Programm erneut. Jede andere Eingabe bringt Sie zum Betriebssystem zurück. Es erscheint wieder das MS-DOS-Prompt.

DISKCOPY

Geben Sie den Zusatz /V beim Programmaufruf mit ein, weisen Sie das Programm an, die kopierten Daten mit den Ursprungsdaten zu vergleichen. In diesem Falle erscheint nach beendetem Kopiervorgang die Meldung:

Verifying...	Überprüfung...

Das Programm DISKCOPY vergleicht jetzt die Daten der Zieldiskette mit den Daten der Ursprungsdiskette. Es wird dieselbe Funktion ausgeführt wie mit dem Programm DISKCOMP. Eine Meldung des Programmes erfolgt nur, wenn Unterschiede festgestellt werden. Es erscheint die Meldung:

Disk verify failure	Diskettenprüffehler

Diese Meldung bezieht sich nur auf fehlerhaft übertragene Daten. Enthält die neue Diskette fehlerhafte Sektoren, kann sie für das Programm DISKCOPY nicht benutzt werden, da der fehlerhafte Sektor nach dem Formatieren immer gesperrt wird. Dies wird durch die Meldung

Disk copy failure	Kopie nicht möglich

angezeigt. Formatieren Sie diese Diskette mit dem Programm FORMAT neu. Sie kann dann immer noch für andere Daten benutzt werden. Wiederholen Sie das Programm DISKCOPY mit einer anderen Zeichenkette.

Rufen Sie das Programm zusammen mit einem Laufwerknamen auf, wird dieses Laufwerk vom Programm als Ursprungslaufwerk angenommen. Sie werden dann nur noch aufgefordert, das Ziellaufwerk einzugeben.

Nach Eingabe von

DISKCOPY A:

erscheint folgende Meldung:

```
            DISKCOPY Version 3.06
Copyright (C) 1985, Zenith Data Systems Corporation

Destination drive name? (A-F):_
```

Nach Eingabe einer weiteren Laufwerkbezeichnung erscheint die Aufforderung:

```
Place the source disk in A and the destination disk in ..
Press RETURN when ready.
```

Nach Betätigen der Taste **RETURN** beginnt der Kopiervorgang.

Rufen Sie das Programm mit zwei Laufwerkbezeichnungen auf, erscheint sofort die Aufforderung, beide Disketten einzulegen. Geben Sie ein

DISKCOPY A: B:

erscheint folgende Meldung:

```
            DISKCOPY Version 3.06
Copyright (C) 1985, Zenith Data Systems Corporation

Place the source sidc in A and the destination disk in B.
Press RETURN when ready.
```

Auch jetzt beginnt der Kopiervorgang nach Betätigen der Taste **RETURN**.

Fehlermeldungen

Cannot DISKCOPY a Network drive — Ein Laufwerk des Netzwerkes kann nicht kopiert werden

ERKLÄRUNG:
Sie haben versucht, mit dem Befehl DISKCOPY auf ein Laufwerk des Netzwerkes zu kopieren. Dies ist nicht möglich.

Disk copy failure — Kopie nicht möglich

ERKLÄRUNG:
Die Zieldiskette ist fehlerhaft, eventuell beschädigt.

Disk verify failure — Diskettenprüffehler

ERKLÄRUNG:
Die Daten sind nicht richtig auf die Zieldiskette kopiert worden. Die Zieldiskette ist eventuell beschädigt.

Drives specified must not be the same — Laufwerkangaben dürfen nicht gleich sein

ERKLÄRUNG:
Sie haben für beide Disketten das gleiche Laufwerk benannt. Dies ist nicht möglich. Geben Sie den Befehl neu ein.

Format failure on destination drive — Zieldiskette kann nicht formatiert werden

ERKLÄRUNG:
Die Zieldiskette ist beschädigt und kann nicht formatiert werden.

Incompatible media Cannot continue	Disketten nicht kompatibel Programm wird abgebrochen

ERKLÄRUNG:
Sie versuchen, die Daten auf eine Platte eines anderen Typs zu kopieren. Die Meldung erscheint z. B., wenn Sie versuchen, Dateien einer Diskette auf eine Winchester zu kopieren oder umgekehrt. Das ist jedoch nur mit dem Befehl COPY möglich.

Read error on destination drive	Lesefehler der Zieldiskette

ERKLÄRUNG:
Die Daten sind nicht richtig kopiert worden. Wiederholen Sie den Vorgang. Wird der Fehler wieder gemeldet, kann die Diskette nicht benutzt werden.

Read error on Source drive	Lesefehler der Ursprungsdiskette

ERKLÄRUNG:
Die Diskette hat fehlerhafte Sektoren oder ist nicht von DOS formatiert worden.

Write error on destination drive	Schreibfehler der Zieldiskette

ERKLÄRUNG:
Die Zieldiskette ist beschädigt oder mit einem Schreibschutz versehen.

ECHO Intern

Zweck

unterdrückt die Ausgabe der Befehlszeilen in einer Batchdatei, zeigt einen Kommentar

Syntax

ECHO [ON¦OFF]
ECHO [Text]

dabei bedeutet:

ON Einschalten der Anzeigefunktion

OFF Ausschalten der Anzeigefunktion

Text auszugebender Text

Bemerkungen

Mit dem Befehl ECHO kann die Anzeige aller Systemmeldungen, mit Ausnahme der Fehlermeldungen, unterdrückt werden. Sie können den Befehl sowohl in einer Batchdatei eingeben, als auch direkt auf Betriebssystemebene.

Geben Sei den Befehl

ECHO

direkt ein, wird der jeweilige Status angezeigt. Dies ist nach dem Systemstart:

ECHO is on ECHO ist eingeschaltet

Dies können Sie ändern durch Eingabe von:

ECHO OFF

Jetzt werden alle Systemmeldungen, einschließlich des Prompt, unterdrückt. Fragen Sie den Status ab, wird angezeigt:

ECHO is off   ECHO ist ausgeschaltet

Beispiel

Sollen die Befehlszeilen einer Batchdatei während der Abarbeitung nicht angezeigt werden, geben Sie den Befehl ECHO OFF zu Anfang der Datei ein. Jetzt werden allerdings auch keine Kommentare mehr angezeigt. Dies gilt sowohl für den Befehl REM, als auch für den Befehl PAUSE. Sollen jetzt Hinweise für den Benutzer angezeigt werden, müssen Sie den Text nach dem Befehl ECHO eingeben. Ein Beispiel:

```
ECHO OFF
...
...
...
ECHO Dies ist ein Kommentar.
```

In diesem Beispiel werden sämtliche Systemanzeigen auf dem Bildschirm unterdrückt (außer Fehlermeldungen), bis auf die Zeile:

```
Dies ist ein Kommentar.
```

Geben Sie in dem o. a. Beispiel in der nächsten Zeile den Befehl PAUSE ein, sieht die Datei folgendermaßen aus:

```
ECHO OFF
...
...
...
ECHO Dies ist ein Kommentar.
PAUSE Legen Sie eine neue Diskette in Laufwerk B.
...
...
```

ECHO

Der Text des Befehls PAUSE wird nicht angezeigt. Die Abarbeitung der Batchdatei wird aber angehalten, und es erscheinen folgende Zeilen:

```
Dies ist ein Kommentar.
Strike a key when ready ...
```

Nach Beendigung der Batchdatei wird die Funktion ECHO automatisch wieder eingeschaltet.

EXIT Intern

Zweck

verläßt das Betriebssystem und kehrt zu einem Anwenderprogramm zurück.

Syntax

```
EXIT
```

Bemerkungen

Mit diesem Befehl verlassen Sie das Betriebssystem und kehren zu Ihrem Anwenderprogramm zurück. Von einigen Anwenderprogrammen aus können Sie weitere Programme aufrufen. Sie können z. B. auch die Datei COMMAND.COM aufrufen und sind damit wieder im Betriebssystem. Es erscheint das normale Prompt von MS-DOS. Nähere Einzelheiten zur Datei COMMAND.COM im Kapitel 9.

Sie können jetzt auf Betriebssystemebene unter Kontrolle des Anwenderprogrammes interne Befehle aufrufen. Wollen Sie das Betriebssystem verlassen und zum Anwenderprogramm zurückkehren, geben Sie ein:

```
EXIT
```

FC Extern

Zweck

vergleicht zwei Dateien und zeigt eventuelle Unterschiede an

Syntax

[d:] [Weg] FC [/X...] Dateibez1 Dateibez2

dabei bedeutet:

d: vor dem Programmnamen bestimmt das aufzurufende Laufwerk

Weg vor dem Programmnamen bestimmt das aufzurufende Verzeichnis

Dateibez1 erste zu vergleichende Datei

Dateibez2 zweite zu vergleichende Datei

/X... Kombination aus folgenden Zusatzeingaben:

- **/a** bedeutet, daß bei unterschiedlichen Textzeilen nur die erste und letzte differierende Zeile gezeigt wird.
- **/b** bedeutet, daß die Dateien Byte für Byte verglichen werden. Nach der ersten Differenz wird keine Gemeinsamkeit mehr gesucht.
- **/c** bedeutet, daß alle Buchstaben einer Textdatei als Großbuchstaben angesehen werden.
- **/l** bedeutet, daß beide Dateien als Textdateien verglichen werden.
- **/lbn** bestimmt die Größe des Zwischenspeichers für Zeilen bei einem Vergleich von Textdateien

/n zeigt Zeilennummern bei einem Vergleich von Textdateien

/t bedeutet, daß Tabulatoren und einfache Leerfelder als verschiedene Zeichen behandelt werden.

/w bedeutet, daß alle Zwischenräume einer Textdatei als gleich groß angesehen werden.

/# bestimmt die Zahl der Zeilen in Textdateien, die nach einer Differenz wieder gleich sein müssen.

Bemerkungen

Das Programm FC kann im *Netzwerk* **ohne** Einschränkungen benutzt werden.

Mit diesem Dienstprogramm lassen sich fast identische Dateien vergleichen. Bearbeiten Sie einen Text oder entwickeln Sie ein Programm, ist es manchmal erforderlich, leichte Unterschiede herauszusuchen, um festzustellen, welche Datei die aktuellere ist.

FC vergleicht die zweite benannte Datei mit der zuerst eingegebenen. Die beiden Dateien müssen dafür nicht im gleichen Laufwerk sein. Festgestellte Differenzen werden auf dem Bildschirm ausgegeben oder bei Datenumleitung zum neubenannten Ausgabegerät. Es wird der Teil der Dateien gezeigt, ab dem sich die erste Datei von der zweiten unterscheidet.

Mit dem Programm FC können Sie Textdateien oder Ursprungsdateien eines Computerprogrammes (erstellt mit einem Textverarbeitungsprogramm) und Binärdateien (von einem Compiler oder Assembler erstellte Dateien) vergleichen. Dabei werden Textdateien normalerweise Zeile für Zeile und Binärdateien Byte für Byte verglichen. Über eine Zusatzeingabe können Sie aber bestimmen, daß auch Textdateien Byte für Byte verglichen werden.

Bei einem zeilenweisen Vergleich werden vom Programm die unterschiedlichen Zeilen jeweils für beide Dateien angezeigt. Bei einem byteweisen Vergleich werden die unterschiedlichen Bytes nebeneinander aufgelistet.

FC

Vergleichen Sie unterschiedliche Textdateien, werden die Unterschiede der Dateien für jede Datei separat angezeigt. Es beginnt mit der ersten Datei. Nach dem Dateinamen erscheinen die Zeilen, die nicht mehr mit der zweiten Datei übereinstimmen. Die letzte ausgegebene Zeile ist die erste, wieder übereinstimmende Zeile. Danach wird der zweite Dateiname angezeigt und ebenfalls die differierenden Zeilen bis zur ersten wieder übereinstimmenden. Über eine Zusatzeingabe können Sie festlegen, wieviel Zeilen nach einer Differenz gleich sein müssen, damit das Programm beide Dateien als gleich anerkennt. Wird dieser Zusatz nicht eingegeben, werden drei Zeilen angenommen. Die Bildschirmausgabe erfolgt folgendermaßen:

---------<Dateibez1>
unterschiedliche Zeilen
erste übereinstimmende Zeile

---------<Dateibez2>
unterschiedliche Zeilen
erste übereinstimmende Zeile

In dieser Form werden alle gefundenen Unterschiede aufgezeigt. Bei zu vielen Unterschieden wird jedoch der Vergleich abgebrochen und es erscheint die Meldung:

Files are Different ***Dateien sind unterschiedlich***

Während des Vergleiches von Textdateien benutzt das Programm den gesamten Arbeitsspeicher. Sind die Dateien zu groß, um beide komplett geladen zu werden, werden nur die im Arbeitsspeicher befindlichen Teile verglichen. Festgestellte Differenzen werden, wie oben beschrieben, angezeigt. Ist der Teilvergleich beendet, wird das Programm abgebrochen. Ein weiteres Nachladen der Dateien erfolgt nicht.

Sind die zu vergleichenden Binärdateien größer als der Arbeitsspeicher, werden auch hier nur Teile der Dateien geladen. Anders als bei Textdateien werden Binärdateien nachgeladen und bis zum Ende komplett verglichen.

Befehlszeile

In der Befehlszeile muß von Ihnen mindestens der Programmname und die genaue Bezeichnung der beiden Dateien eingegeben werden. Die Zusatzeingaben, mit denen Sie den Programmablauf beeinflussen können, müssen sofort nach dem Programmnamen eingegeben werden. Jede Eingabe nach der zweiten Dateibezeichnung wird ignoriert. Sind die Dateien nicht im Arbeitslaufwerk gespeichert, so können vor dem Dateinamen das Laufwerk und/ oder der Name des Verzeichnisses angegeben werden. Jedes Eingabefeld muß durch eine Leertaste getrennt werden. Ein Beispiel:

```
FC /b C:FOOALPHA.ASM A:BETA.ASM
```

Byte für Byte wird die Datei ALPHA.ASM, die im Verzeichnis FOO des Laufwerkes C: ist, mit der Datei BETA.ASM, die im Laufwerk A: ist, verglichen. Die Unterschiede werden auf dem Bildschirm angezeigt.

Zusatzeingaben

Mit den Zusatzeingaben können Sie bestimmte Funktionen des Programmes aufrufen. Diese Eingaben bestehen aus einem einzelnen Buchstaben, dem ein Schrägstrich vorangestellt werden muß. Der Buchstabe muß immer als **Kleinbuchstabe** eingegeben werden.

Die Eingabe muß nach dem Programmnamen erfolgen. An anderer Stelle wird sie ignoriert. Bei einer Falscheingabe erscheint eine kurze Übersicht mit gültigen Zusatzeingaben. Ohne weitere Eingaben vergleicht das Programm die beiden Dateien folgendermaßen:

- Zeile für Zeile bei Textdateien
- Byte für Byte bei Binärdateien
- Groß- und Kleinbuchstaben werden berücksichtigt
- unterschiedliche Zwischenräume werden berücksichtigt
- nach einem Unterschied müssen in Textdateien mindestens drei Zeilen wieder übereinstimmen

/a -- ASCII verkürzte Ausgabe

Mit diesem Zusatz wird bei einem zeilenweisen Vergleich von Textdateien die Ausgabe verkürzt. FC zeigt nur noch die erste und die letzte unterschiedliche Zeile.

/b -- Binary

Mit diesem Zusatz werden Textdateien immer byteweise verglichen. Dabei werden nach einer Differenz keine Gemeinsamkeiten mehr gesucht. Die Ausgabe der Unterschiede erfolgt in folgender Form:

```
--ADDRS----Dateibez1----Dateibez2

xxxxxxxx    yy          zz
```

dabei bedeutet:

xxxxxxxx	relative Adresse ab Dateianfang
yy	differierendes Byte der ersten Datei
zz	differierendes Byte der zweiten Datei

Ist eine der beiden Dateien größer, und es können am Ende nicht alle Bytes verglichen werden, erscheint die Meldung:

Data left in ...   ***noch Daten in Datei...***

Dieser Zusatz darf nicht mit dem Zusatz /l eingegeben werden.

/c -- Buchstabengröße nicht berücksichtigen

Dieser Zusatz gilt für Textdateien. Es werden alle Wörter als gleich angesehen, unabhängig davon, ob sie mit Groß- oder Kleinbuchstaben geschrieben wurden. Der Text wird vom Programm in Großbuchstaben umgeformt. Im folgenden Beispiel sind beide Zeilen gleich:

Beide ZEILEN sind JETZT gleich

Beide Zeilen sind jetzt gleich

Dieser Zusatz kann auch mit dem Zusatz /w zusammen benutzt werden.

/l -- Vergleich zeilenweise

Dieser Zusatz gilt für Textdateien. Es werden immer komplette Zeilen verglichen. Achten Sie darauf, den Zwischenspeicher für unterschiedliche Zeilen entsprechend zu bestimmen. Ohne weitere Eingabe kann der Speicher maximal 100 Zeilen aufnehmen. Dieser Zusatz darf nicht mit dem Zusatz /b eingegeben werden.

/lb*n* -- Zeilenspeicher bestimmen

Dieser Zusatz gilt für Textdateien. Er bestimmt die Größe des Zwischenspeichers für Textzeilen. Ohne weitere Eingabe kann dieser Speicher 100 Zeilen aufnehmen. Vergleichen Sie Dateien, die sehr stark differieren, sollten Sie den Speicher vergrößern. Es muß genügend Platz für die maximale Zahl der zusammenhängend differierenden Zeilen vorhanden sein.

/n -- zeige Zeilennummer

Mit diesem Zusatz weisen Sie FC an, vor jede Zeile, die auf dem Bildschirm gezeigt wird, eine *Zeilennummer* anzuzeigen. Die zu durchsuchenden Dateien werden dabei, immer mit Null beginnend, durchnumeriert. Das ist z. B. wünschenswert, wenn Sie die Zeilen anschließend ändern wollen. Sie können so die gewünschte Stelle mit einem Textverarbeitungsprogramm schnell auffinden.

/t -- Tabulator ist nicht Leerzeichen

Dieser Zusatz gilt für Textdateien. Es werden Zwischenräume mit den Zeichen für Tabulator oder einfachem Leerfeld als Unterschied angesehen. Geben Sie den Zusatz nicht ein, sind diese Zwischenräume für FC immer gleich.

/w -- Zwischenräume egalisieren

Dieser Zusatz gilt für Textdateien. Es werden alle Zwischenräume als gleich angesehen, unabhängig davon, ob es ein einzelnes Feld ist, oder ob es mehrere Felder sind. Die Zwischenräume werden vom Programm auf je ein Feld reduziert. Sie werden immer mit berücksichtigt. Im folgenden Beispiel sind beide Zeilen gleich:

Beide_Zeilen_sind_jetzt_gleich

Beide___Zeilen_________sind______jetzt__gleich

Benutzen Sie die Zusätze /w und /c zusammen, sind die beiden folgenden Zeilen gleich:

Beide_ZEILEN_sind_JETZT_gleich

Beide___Zeilen_______sind_____jetzt___gleich

/# -- Zahl der gleichen Zeilen

Dieser Zusatz gilt für Textdateien. Er bestimmt die Zahl der Zeilen, die nach einer Differenz wieder gleich sein müssen, damit die beiden Dateien gleich sind. Ohne Eingabe des Zusatzes werden vom Programm drei Zeilen angenommen. Sie können hier zwischen 1 und 9 Zeilen bestimmen.

Beispiel 1

Die beiden folgenden Dateien sollen miteinander verglichen werden:

ALPHA.ASM	BETA.ASM
DATEIA	DATEIB
A	A
B	B
C	C
D	G
E	H
F	I
G	J
H	1
I	2
M	P
N	Q
O	R
P	S
Q	T
R	U
S	V
T	4
U	5
V	W
W	X
X	Y
Y	Z
Z	

Geben Sie ein:

FC ALPHA.ASM BETA.ASM

Die Dateien werden als Textdateien verglichen. Für eine erneute Übereinstimmung müssen drei Zeilen gleich sein. Die Ausgabe der Differenzen erfolgt auf dem Bildschirm. Es werden folgende Differenzen angezeigt:

```
---------ALPHA.ASM
D
E                                        (ALPHA.ASM enthält DEFG,
F                                        wo BETA.ASM nur G ent-
G                                        hält)

---------BETA.ASM
G
```

```
---------ALPHA.ASM
M
N                                        (ALPHA.ASM enthält MNOP,
O                                        wo BETA.ASM J12P ent-
P                                        hält)

---------BETA.ASM
J
1
2
P
```

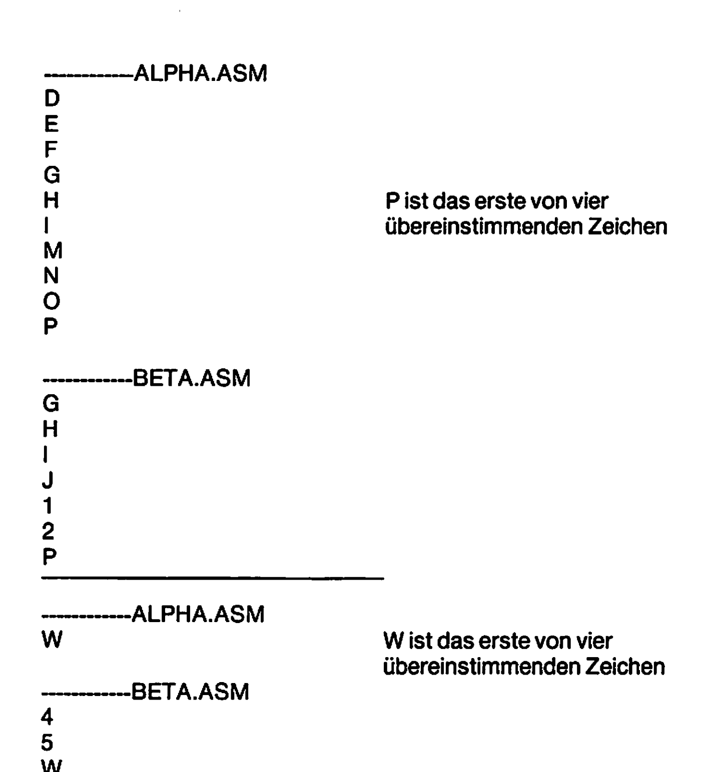
```
---------ALPHA.ASM
W                                        (ALPHA.ASM enthält W, wo
                                         BETA.ASM 45W enthält)
---------BETA.ASM
4
5
W
```

Beispiel 2

Geben Sie den Befehl für dieselben Dateien leicht verändert ein, so daß jetzt vier Zeilen übereinstimmen müssen und die Unterschiede auf den Drucker ausgegeben werden:

```
FC /4 ALPHA.ASM BETA.ASM > PRN
```

Es sollte folgendes gedruckt werden:



```
----------ALPHA.ASM
D
E
F
G
H
I
M
N
O
P

-----------BETA.ASM
G
H
I
J
1
2
P
```

P ist das erste von vier übereinstimmenden Zeichen

```
-----------ALPHA.ASM
W

-----------BETA.ASM
4
5
W
```

W ist das erste von vier übereinstimmenden Zeichen

Beispiel 3

Für das folgende Beispiel werden wieder die beiden vorhandenen Dateien verglichen. Diesmal wird der Zusatz /B mit eingegeben. Die Ausgabe auf dem Bildschirm soll seitenweise mit dem Filter MORE erfolgen. Geben Sie ein:

```
FC /b ALPHA.ASM BETA.ASMMORE
```

Es erscheint auf dem Bildschirm folgendes:

```
--ADDRS------F1----F2--
00000009    44   47
0000000C    45   48
0000000F    46   49
00000012    47   4A
00000015    48   31
00000018    49   32
0000001B    4D   50
0000001E    4E   51
00000021    4F   52
00000024    50   53
00000027    51   54
0000002A    52   55
0000002D    53   56
00000030    54   34
00000033    55   35
00000036    56   57
00000039    57   58
0000003C    58   59
0000003F    59   5A
00000042    5A   1A
```

Weitere Erklärungen

Genauso wie sich der Befehl mit einem Filter verketten läßt, ist es möglich, die Ausgabe der Differenzen in eine neue Datei einzugeben. Damit lassen sich zwei größere Dateien leichter vergleichen. (Nähere Einzelheiten zur Datenumleitung finden Sie im Kapitel 8.) Ein Beispiel:

FC DATEI1 DATEI2 > DIFFERENZ

Es werden jetzt die beiden Dateien verglichen. Die Differenzen werden in eine neue Datei mit dem Namen DIFFERENZ gegeben.

Fehlermeldungen

Bad file, Read error in	Fehlerhafte Datei, Lesefehler in Datei

ERKLÄRUNG:
Das Programm kann die Datei nicht komplett lesen. Überprüfen Sie die Diskette mit CHKDSK.

File not found	Datei nicht gefunden

ERKLÄRUNG:
Die eingegebene Datei konnte nicht gelesen werden. Überprüfen Sie Ihre Eingabe, eventuell ist die Datei nicht auf der Platte.

Incorrect DOS version	Falsche DOS-Version

ERKLÄRUNG:
Sie benutzen das Programm FC unter einer DOS-Version, die nicht 2.0 oder höher ist.

Invalid number of parameters	Falsche Anzahl von Zusatzeingaben

ERKLÄRUNG:
Sie haben die Zusatzeingaben falsch eingegeben. Überprüfen Sie Ihre Eingaben.

Invalid parameter X	Falsche Zusatzeingabe X

ERKLÄRUNG:
Die mit X bezeichnete Zusatzeingabe ist falsch. Wiederholen Sie die Eingabe.

FIND Extern

Zweck

sucht in einer oder mehrerer Dateien nach dem bezeichneten Text

Syntax

[d:] [Weg] FIND /X "Zeichenkette" Dateibez ...

dabei bedeutet:

d: vor dem Programmnamen bestimmt das aufzurufende Laufwerk

Weg vor dem Programmnamen bestimmt das aufzurufende Verzeichnis

Zeichenkette Text, nach dem die Dateien abgesucht werden sollen

Dateibez einzelne oder mehrere Dateien, in denen gesucht werden soll

/X Kombination aus den folgenden Zusatzeingaben:

- **/C** zähle Zeilen (es wird nur die Gesamtzahl der Zeilen angezeigt, die die Zeichenkette enthalten)
- **/I** ignoriere Buchstabengröße
- **/N** zeige Zeilennummer (vor jeder gefundenen Zeile wird die entsprechende Zeilennummer angezeigt)
- **/V** zeige andere Zeilen (es werden alle Zeilen der bezeichneten Dateien gezeigt, die die gesuchte Zeichenkette nicht enthalten)

Bemerkungen

Das Programm FIND kann im Netzwerk ohne Einschränkungen benutzt werden.

Mit dem *Filter* FIND können Sie in verschiedenen Dateien nach einem bestimmten Text suchen. Als zu suchende Zeichenkette können Sie jeden beliebigen Text eingeben. Die Zeichen der Zeichenkette müssen in Anführungsstrichen eingeschlossen werden. Der Filter zeigt anschließend jede Zeile der Datei, die die gesuchte Zeichenkette enthält, komplett an. Durch verschiedene Zusatzeingaben können Sie die Arbeitsweise des Filters beeinflussen.

Filter eignen sich besonders zur Befehlsverkettung. Nähere Einzelheiten dazu finden Sie im Kapitel 8.

Zeichenkette

Sie rufen den Filter FIND zusammen mit der zu suchenden *Zeichenkette* auf. Die Zeichenkette muß immer in Anführungsstriche eingeschlossen werden. Sie darf jedes beliebige Zeichen enthalten. Die Länge der Zeichenkette wird nur begrenzt durch den Tastatureingabespeicher. Die gesamte Befehlszeile des Filters FIND darf beim Aufruf maximal 127 Zeichen (plus **RETURN** = 128) umfassen. Geben Sie Zusätze mit ein, müssen diese Zusätze direkt nach dem Befehlsnamen eingegeben werden. Jedem Zusatz wird ein Schrägstrich vorangestellt. Die folgende Zeichenkette wird durch eine Leertaste vom Zusatz getrennt. Geben Sie keine Zusätze ein, wird die Zeichenkette, durch eine Leertaste getrennt, direkt nach dem Befehlsnamen eingegeben.

Enthält die zu suchende Zeichenkette selber Anführungsstriche, muß sie in doppelte Anführungsstriche eingeschlossen werden. Achten Sie bei der Eingabe darauf, daß Sie den Text exakt so eingeben, wie er in den zu durchsuchenden Dateien gespeichert ist. Dies gilt auch für die Groß- und Kleinschreibung (außer beim Zusatz /I) und für die Zwischenräume. Der Filter zeigt nur die Textzeilen an, die exakt die eingegebene Zeichenkette enthalten.

Dateibezeichnung

Geben Sie den genauen Namen der *Datei* ein, die abgesucht werden soll. Der Dateiname wird von der Zeichenkette durch eine Leertaste getrennt. Soll in mehreren Dateien nacheinander gesucht werden, geben Sie die genauen Dateinamen jeweils durch eine Leertaste getrennt ein. Es muß jede Datei genau bezeichnet werden. Die Dateinamen dürfen nicht durch Globalzeichen zusammengefaßt werden. Befindet sich die Datei nicht im momentanen Verzeichnis, stellen Sie den Laufwerknamen und/oder den Namen eines Verzeichnisses voran. Sie können beliebig viele Dateinamen spezifizieren, begrenzt nur durch die Länge der Eingabezeile mit maximal 127 Zeichen.

Zusatzeingaben

Die *Zusatzeingaben* bestimmen die Arbeitsweise des Filters FIND. Sie können je nach Wunsch eingegeben werden. Die Zusatzeingaben müssen direkt nach dem Befehlsnamen eingegeben werden. Am Ende der Befehlszeile werden sie ignoriert. Die Zusätze werden von den anderen Eingaben der Befehlszeile durch Leertasten getrennt. Jedem Zusatz muß ein Schrägstrich (/) vorangestellt werden. Es ist möglich, mehrere Zusätze gleichzeitig einzugeben. Jedoch dürfen Sie die Zusätze /C und /N nicht gleichzeitig eingeben. Der Zusatz /N würde dann nicht beachtet. Geben Sie mehrere Zusätze gleichzeitig ein, trennen Sie die einzelnen Zusätze nur durch den Schrägstrich (/), andere Zeichen (z. B. Leertaste) sind nicht erlaubt.

/C - zähle Zeilen

Normalerweise zeigt FIND jede Zeile auf dem Bildschirm an, die die gesuchte Zeichenkette enthält. Geben Sie den Zusatz /C ein, werden nur die Zeilen gezählt, die die Zeichenkette enthalten. Danach zeigt FIND für jede Datei, die Sie in der Befehlszeile benannt haben, die Anzahl der Zeilen an. Das geschieht in der Form:

```
----------Dateibez: nn
----------Dateibez: nn
...
...
```

nn ist die Anzahl der Zeilen, die die gesuchte Zeichenkette enthalten.

/I - ignoriere Buchstabengröße

Normalerweise findet FIND die bezeichnete Zeichenkette in der Datei nur, wenn eine totale Übereinstimmung zwischen dem Text der Zeichenkette und dem Text der Datei besteht. Durch den Zusatz /I wird diese Bestimmung teilweise aufgehoben. Jetzt werden alle bezeichneten Buchstaben in *Großbuchstaben* umgeformt. Im folgenden Beispiel sind beide Zeilen gleich:

Beide ZEILEN sind JETZT gleich.

Beide Zeilen sind jetzt gleich.

/N - zeige Zeilennummer

Mit diesem Zusatz weisen Sie FIND an, vor jede Zeile, die die gesuchte Zeichenkette enthält, eine *Zeilennummer* anzuzeigen. Die Zeilennummer wird immer in eckige Klammern ([ ]) eingeschlossen. Die zu durchsuchenden Dateien werden dabei, immer mit Null beginnend, durchnumeriert. Findet das Programm die von Ihnen eingegebene Zeichenkette, zeigt es die komplette Zeile mit der dazugehörigen relativen *Zeilennummer* an. Das ist z. B. wünschenswert, wenn Sie die Zeile anschließend ändern wollen. Sie können so die gewünschte Stelle mit einem Textverarbeitungsprogramm schnell auffinden.

/V - zeige andere Zeilen

Normalerweise zeigt FIND die Zeilen, die den gesuchten Text enthalten. Der Zusatz /V bewirkt das Gegenteil. Jetzt werden alle Zeilen angezeigt, die die gesuchte Zeichenkette nicht enthalten.

Beispiel 1

Im Laufwerk B: befindet sich eine Diskette, die folgende Dateien enthält:

KAPITEL1, KAPITEL2, KAPITEL3

FIND

In diesen Textdateien sollen alle Zeilen mit dem Text *„Zenith PC“* herausgesucht werden. Es ist möglich, daß die Buchstaben nicht alle mit Großbuchstaben geschrieben wurden. Daher soll der Zusatz /I mit eingegeben werden. Außerdem wird noch der Zusatz /N eingegeben. Rufen Sie den Filter folgendermaßen auf:

```
FIND /N /I "Zenith PC" KAPITEL1  KAPITEL2  KAPITEL3
```

Der Filter sucht jetzt die Dateien in der eingegebenen Reihenfolge ab und schreibt laufend die gefundenen Zeilen auf den Bildschirm. Dies geschieht in der Form:

```
---------B:KAPITEL1
[15] Benutzer des Zenith PC, die noch keine Erfahrung mit Mikrocom-
[41] Die sichtbaren Teile Ihres Zenith PC bestehen aus einer Zentral-
[54] antwortet Ihr Zenith PC dann in aller Regel. Damit der Informa-
...
...
...

---------B:KAPITEL2
[34] wurde. Der Zenith PC wird werkseitig auf automatischen Systemstart

---------B:KAPITEL3
[23] Außer Ihrem Zenith PC benötigen Sie dafür die drei mitgelieferten
```

Zuerst wird der Name der Datei, die gerade durchsucht wird, geschrieben. Danach werden die Zeilen mit dem bezeichneten Text angezeigt. Im ersten Kapitel sind dies mehrere Zeilen, die Zeilen 15, 41 und 54 sind hier noch einmal wiedergegeben. In der zweiten Datei (Kapitel 2) ist nur eine Zeile vorhanden (Zeile 34), ebenso in der dritten Datei. Findet FIND den gesuchten Text nicht, wird der Dateiname alleine angezeigt. Eine weitere Meldung wird dann nicht mehr ausgegeben.

Beispiel 2

Sie können den Filter auch mit anderen Befehlen oder weiteren Filtern verketten. Z. B. können Sie bestimmte Dateinamen des Inhaltsverzeichnisses nicht anzeigen lassen, im Gegensatz zum Befehl DIR, der bestimmte Dateien anzeigt. Sollen z. B. alle Dateien des Verzeichnisses angezeigt werden, deren Namenszusatz nicht .DAT ist, ist diese Selektion nur mit dem Filter FIND möglich. Geben Sie ein:

DIR ¦ FIND /V "DAT"

Die Ausgabe der Dateinamen, die der Befehl DIR anzeigen würde, wird jetzt in den Filter FIND eingelesen. Es werden alle Dateien mit dem Zusatz „.DAT" herausgenommen. Anschließend werden die restlichen Dateien angezeigt. Bei einer großen Zahl von Dateien kann auch diese Ausgabe noch weiter beeinflußt werden. Geben Sie die Ausgabe von FIND (die selektierten Dateien) in den Filter MORE, erfolgt die Ausgabe seitenweise. Der Befehl lautet dann:

DIR ¦ FIND /V "DAT" ¦ MORE

Ebenso können die ausgewählten Dateien vor der Ausgabe noch sortiert werden. Ein Beispiel:

DIR ¦ FIND /V "DAT" ¦ SORT ¦ MORE

Fehlermeldungen

Incorrect DOS version	Falsche DOS Version

ERKLÄRUNG:
Sie benutzen den Filter FIND unter einer DOS-Version, die nicht 2.0 oder höher ist.

FIND: File not found	FIND: Datei nicht vorhanden

ERKLÄRUNG:
Die eingegebene Datei ist nicht vorhanden. Überprüfen Sie Ihre Eingabe.

FIND

FIND:
Invalid number of parameters

FIND:
Falsche Anzahl der Zusatzeingaben

ERKLÄRUNG:
Sie haben die zu suchende Zeichenkette nicht eingegeben. Überprüfen Sie Ihre Eingabe.

FIND:
Invalid parameter X

FIND:
Falsche Zusatzeingabe X

ERKLÄRUNG:
Sie haben einen falschen Zusatz eingegeben. Der Befehl wird korrekt abgearbeitet, der falsche Zusatz wird dabei ignoriert.

FIND:
Read error in

FIND:
Lesefehler der Datei

ERKLÄRUNG:
Die von Ihnen bezeichnete Datei kann vom Filter FIND nicht richtig gelesen werden. Geben Sie den Befehl neu ein. Tritt der Fehler erneut auf, überprüfen Sie die Diskette mit CHKDSK.

FIND:
Syntax error

FIND:
Schreibfehler

ERKLKÄRUNG:
Sie haben die zu suchende Zeichenkette falsch eingegeben. Überprüfen Sie Ihre Eingabe. Die Zeichenkette muß durch Anführungsstriche eingeschlossen werden.

FOR Intern

Zweck

wiederholt mehrmals einen MS-DOS-Befehl in einer Batchdatei

Syntax

Direkteingabe: **FOR %Variable IN (Wert) DO Befehl**
Batchdatei: **FOR %%Variable IN (Wert) DO Befehl**

dabei bedeutet:

FOR	Befehlsname
Variable	Name der Variablen darf irgendein Zeichen sein außer den Ziffern 0 bis 9 (allgemeine Batchvariable)
IN	Teil des Befehls, der auf Wert hinweist
Wert	wird vom Befehl bearbeitet und kann sein: Dateibez Verzeichnis Dateiname Zeichenkette
DO	Teil des Befehls, der auf den auszuführenden Befehl hinweist
Befehl	Befehl, der mehrfach ausgeführt werden soll

Bemerkungen

Der Befehl FOR kann sowohl direkt als auch über eine Batchdatei eingegeben werden. In der Direkteingabe wird zur Kennzeichnung der *Variablen* das Prozentzeichen (%) nur einmal eingegeben. In einer Batchdatei muß es zweimal eingegeben werden, damit die Variable in dieser Befehlszeile von den allgemeinen Variablen unterschieden wird.

FOR

Der Befehl FOR ist folgendermaßen aufgebaut:

Der in der Befehlszeile aufgeführte Befehl wird mit jedem *Wert*, der in der Klammer benannt ist, ausgeführt. Die Variable reicht jeden *Wert* weiter an den *Befehl.*

Der auszuführende Befehl kann jeder beliebige MS-DOS-Befehl sein. Die Variable darf mit jedem belieben Zeichen definiert werden, außer mit Ziffern. Mit *Wert* dürfen beliebige Dateinamen oder andere Worte bezeichnet werden. Dateinamen dürfen auch mit den Globalzeichen zusammengefaßt werden. Der Wert ist immer in runde Klammern einzuschließen. Die Worte **FOR, IN** und **DO** müssen unbedingt mit Großbuchstaben eingegeben werden, andernfalls werden sie nicht richtig vom Befehlsinterpreter erkannt. Es erscheint dann die Fehlermeldung:

Syntax Error	Schreibfehler

Ein Beispiel:

```
FOR %%A IN (*.BAT) DO TYPE %%A > PRN
```

Diese Befehlszeile wird in einer Batchdatei folgendermaßen abgearbeitet:

Im ersten Teil des Befehls wird die Variable mit A definiert. Dann werden sämtliche Dateien des Inhaltsverzeichnisses mit dem Namenszusatz .BAT herausgesucht. Jeder einzelne Dateiname wird für die Variable %A an den Befehl weitergegeben. Dieser wird dann entsprechend ausgeführt. In diesem Fall werden die Inhalte sämtlicher Batchdateien auf den Drucker ausgegeben. Die Befehlszeile könnte dann folgendermaßen aussehen:

```
TYPE AUTOEXEC.BAT > PRN
```

Geben Sie für Wert mehrere verschiedene Namen ein, müssen diese durch eine Leertaste getrennt werden.

In einer Batchdatei ist es unbedingt erforderlich, die Variable mit zwei Prozentzeichen zu definieren. Der Prozessor erkennt am Prozentzeichen die Variablen. Ein Zeichen definiert die normale Batchvariable, die durch die Ziffern 0 bis 9 bezeichnet werden. Da Sie die Variable des Befehls FOR mit einem anderen Zeichen definiert haben, würde der Prozessor jetzt eine Fehlermeldung ausgeben und den Befehl übergehen. Bei zwei Zeichen nimmt der Prozessor das erste Prozentzeichen als normale Variable und erkennt dann am zweiten Prozentzeichen, daß es sich um die Variable des Befehls FOR handelt.

Fehlermeldungen

Invalid number of parameters — Falsche Anzahl der Zusatzeingaben

ERKLÄRUNG:
Sie haben nicht alle Teile des Befehls FOR eingegeben oder versucht, mehr als einen FOR-Befehl in einer Zeile einzugeben. Überprüfen Sie Ihre Eingabe.

Syntax Error — Schreibfehler

ERKLÄRUNG:
Sie haben den Befehl FOR nicht richtig geschrieben. Überprüfen Sie Ihre Eingabe.

FOR cannot be nested — FOR kann nicht verschachtelt werden

ERKLÄRUNG:
Sie rufen mit einem FOR-Befehl einen weiteren FOR-Befehl auf. Damit würde innerhalb einer FOR-Schleife eine weitere Schleife gebildet werden. Dies ist auf Betriebssystemebene nicht möglich.

FORMAT Extern

Zweck

bereitet eine Platte zur Aufnahme von MS-DOS-Dateien vor

Syntax

[d:] [Weg] FORMAT [D:] [/X...]

dabei bedeutet:

d: vor dem Programmnamen bestimmt das aufzurufende Laufwerk

Weg vor dem Programmnamen bestimmt das aufzurufende Verzeichnis

D: Name des Laufwerkes, in dem die zu formatierende Platte ist

/X... Kombination aus folgenden Zusatzeingaben:

- **/B** bedeutet, daß Platz für die drei Systemdateien reserviert wird. Anschließend kann jede MS-DOS-Version auf die Platte kopiert werden.
- **/C** bedeutet, daß die Platte nicht initialisiert wird. Es wird nur das Rootdir und das FAT gelöscht. Wurde die Diskette mit dem Zusatz /M formatiert, muß dieser Zusatz mit eingegeben werden.
- **/4** bedeutet, daß in einem High-Density-Laufwerk Disketten mit einer Kapazität von 360 KBytes formatiert werden.
- **/8** bedeutet, daß nur 8 Sektoren der Diskette benutzt werden. Es werden immer 9 Sektoren formatiert.
- **/M** bedeutet, daß nur eine Seite der Diskette formatiert wird.

/N bedeutet, daß alle Anfragen und Erklärungen unterdrückt werden (nützlich in einer Batchdatei).

/S bedeutet, daß die Systemdateien auf die formatierte Platte kopiert werden.

/V bedeutet, daß die Platte nach dem Formatieren auf Fehler überprüft wird.

Bemerkungen

Das Programm FORMAT kann im *Netzwerk* **nicht** benutzt werden.

Format ist das erste Programm, mit dem Sie eine neue Platte bearbeiten. Das Programm bereitet sowohl Winchesterlaufwerke als auch Disketten zur Datenaufnahme vor. Es schreibt dabei die *Grundinformationen* für den Schreib-Lese-Kopf auf die Platte. Damit kann dieser dann auf die gewünschten Daten korrekt zugreifen. Sie können nur mit Platten arbeiten, die vorher formatiert wurden.

Durch Zusatzeingaben können Sie die Diskette als mit 8 Sektoren formatiert kennzeichnen (es werden aber immer 9 Sektoren pro Spur geschrieben) oder nur einseitig formatieren. Das Programm FORMAT kopiert immer das Programm BOOT-LOADER auf den ersten Sektor der ersten Seite in die Spur 0 (absolute Adresse: Seite 0, Spur 0, Sektor 1). Wird der Zusatz /S eingegeben, werden zusätzlich die Systemdateien in der Reihenfolge IO.SYS, MSDOS.SYS, COMMAND.COM auf die Platte kopiert.

Ihr Zenith PC arbeitet mit 5.25-Inch-Disketten. Diese Disketten müssen für normale Laufwerke eine Mindestqualität von 48-tpi (Tracks per Inch) haben und werden mit 40 Spuren (0 bis 39) auf jeder Seite formatiert. Für High-Density-Laufwerke benötigen Sie 5.25-Inch-Disketten mit einer Mindestqualität von 96-tpi. Hier werden 80 Spuren je Seite formatiert.

Das folgende Bild zeigt die Aufteilung der Diskette in Spuren und Bereiche.

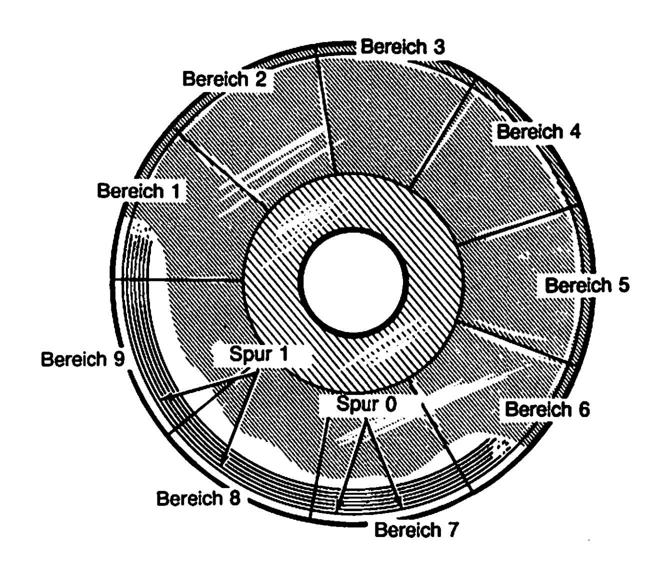


Bild 11.2
Diskettenaufteilung

Tabelle 11.3 Diskettenformate

DISKETTE	TPI	SPUREN	SEKTOREN	KAPAZITÄT
Single-Sided	48	40	8/9	160 - 180K
Double-Sided	48	40	8/9	320 - 360K
High-Capacity	96	80	15	1MByte

ACHTUNG:
Sie können dieses Programm nicht aufrufen, wenn Sie mit dem Dienstprogramm ASSIGN Laufwerkanfragen vorübergehend umgeleitet haben. Dies gilt auch dann, wenn der Name des zu formatierenden Laufwerkes nicht geändert wurde. Löschen Sie dann erst die vorgenommenen Änderungen (s. a. Seite 11.6).

Rufen Sie das Programm ohne Zusatzeingaben auf, wird die Diskette folgendermaßen formatiert:

- Doppelseitig (DS)
- 9 Sektoren pro Spur (für normale Laufwerke) oder
- 15 Sektoren pro Spur (für High-Density-Laufwerke)
- Keine Kopie der Systemdateien (nur BOOT-LOADER)
- Alle Anfragen werden angezeigt

Geben Sie ein:

```
FORMAT
```

und es erscheint auf dem Bildschirm folgendes:

```
            FORMAT Version 3.05
Copyright (C) 1985, Zenith Data Systems Corporation

Drive to format? _:
```

Geben Sie jetzt den Buchstaben für das Laufwerk mit der zu formatierenden Diskette ein, z. B. B. Es erscheint folgende Meldung:

Insert new disk in drive B and press RETURN when ready.	Legen Sie eine neue Diskette in das Laufwerk B und betätigen Sie RETURN wenn Sie fertig sind.

Nach Betätigen der Taste **RETURN** wird die Diskette formatiert. Sollten auf einer schon benutzten Diskette Daten gespeichert sein, werden diese jetzt gelöscht.

Geben Sie in der Befehlszeile das Laufwerk gleich mit ein, erscheint z. B. folgende Meldung:

```
                    FORMAT Version 3.05
Copyright (C) 1985, Zenith Data Systems Corporation

Insert new disk in drive B
and press RETURN when ready.
```

Zusatzeingaben

Nach der Laufwerkbezeichnung können Sie mit *Zusatzeingaben* bestimmte Funktionen aufrufen. Jede Funktion wird durch einen einzelnen Buchstaben oder eine Ziffer, dem/der ein Schrägstrich vorangeht, bestimmt. Es ist möglich, mehrere Funktionen zusammen aufzurufen.

/B -- reserviere Platz für Systemdateien

Mit diesem Zusatz reserviert FORMAT auf der neuen Platte Speicherplatz für die drei Systemdateien: IO.SYS, MSDOS.SYS und COMMAND.COM. Die beiden ersten Dateien müssen an bestimmten Stellen auf der Platte vorhanden sein, damit sie bootfähig ist. Benutzen Sie diesen Zusatz, können Sie nach der Formatierung bereits Daten auf die Platte kopieren und erst später ein Betriebssystem.

/C -- lösche FAT und Inhaltsverzeichnis

Bei Eingabe dieses Zusatzes wird die Platte nicht initialisiert. Es wird nur das Rootdir und die Dateizuordnungstabelle (FAT) gelöscht. Wurde die Diskette mit dem Zusatz /M formatiert, muß dieser Zusatz wieder mit eingegeben werden.

/8 -- 8 Sektoren pro Spur

Bei Eingabe dieses Zusatzes werden nur 8 Sektoren der Diskette benutzt (entspricht der Version 1.25). Es werden aber immer 9 Sektoren formatiert.

/4 -- 40 Spuren je Seite

Dieser Zusatz kann nur für High-Density-Laufwerke benutzt werden. Das Programm formatiert die Disketten nur mit 40 Spuren je Seite. Dies ergibt eine Kapazität von 360 KBytes. Damit entspricht dies dem Format der Version 2. So formatierte Disketten können auf einem bisherigen Laufwerk jedoch nur gelesen, aber nicht bespielt werden.

/M -- formatiere einseitig

Dieser Zusatz bestimmt, daß nur eine Seite der Diskette formatiert wird. Es können dann auch nur auf einer Seite Daten gespeichert werden.

/N -- zeige keine Meldungen

Dieser Zusatz bestimmt, daß alle Anfragen und Statusmeldungen unterdrückt werden. Diese Funktion kann in einer Batchdatei sehr nützlich sein.

/S -- Systemdateien kopieren

Dieser Zusatz bestimmt, daß die Systemdateien auf die formatierte Platte kopiert werden. Es sind dies die Dateien:

IO.SYS
MSDOS.SYS
COMMAND.COM

/V -- überprüfe auf fehlerhafte Sektoren

Dieser Zusatz bestimmt, daß die Platte nach dem Formatieren auf fehlerhafte Sektoren überprüft wird. Diese Sektoren werden im FAT eingetragen und können damit nicht mehr benutzt werden. Dieser Zusatz gilt nur für die Diskette. Die Winchester wurde bereits mit dem Programm PREP überprüft. In der Statusmeldung werden die fehlerhaften Sektoren angezeigt.

Meldungen

Haben Sie nach Aufruf des Programmes die Laufwerkbezeichnung eingegeben, erscheint für das Formatieren einer Diskette folgende Aufforderung:

Insert new disk in drive X and press RETURN when ready.	Legen Sie neue Diskette in Lauf- werk X und Betätigen Sie RETURN.

Formatieren Sie den Teil eines Winchesterlaufwerkes, erscheint eine ähnliche Meldung:

Will format partition assigned drive X Press RETURN when ready.	Es wird der Teil, der mit X bezeichnet wurde, formatiert. Betätigen Sie RETURN.

Haben Sie den Zusatz /S mit eingegeben, werden nach dem Formatieren die Systemdateien kopiert. Ist auch dieser Vorgang beendet, erscheint die Mitteilung:

System transferred	System übertragen

Danach werden Sie aufgefordert, einen Namen für die Platte einzugeben oder die Anfrage mit **RETURN** zu übergehen:

Enter desired volume label (11 characters, RETURN for none)?	Geben Sie den gewünschten Namen ein (11 Zeichen, oder RETURN wenn nicht gewünscht)?

Für den Plattennamen gelten dieselben Bestimmungen wie für Dateinamen. Sie können diesen Namen jederzeit mit dem Programm LABEL verändern oder löschen.

Nach dem Formatiervorgang wird eine *Statusmeldung* der Diskette angezeigt. Sie ersehen daraus die formatierte Kapazität und die zur freien Verfügung stehende Kapazität. Es erscheinen folgende Meldungen:

Wurde kein Zusatz mit eingegeben (für 48-tpi-Disketten):

362496 bytes total disk space	362496 Bytes Gesamtkapazität
362496 bytes availabe on disk	362496 Bytes freie Kapazität

Wurde kein Zusatz mit eingegeben (für 96-tpi-Disketten):

1213952 bytes total disk space	1213952 Bytes Gesamtkapazität
1213952 bytes availabe on disk	1213952 Bytes freie Kapazität

Wurde der Zusatz /S mit eingegeben:

362496 bytes total disk space	362496 Bytes Gesamtkapazität
45568 bytes used by system	45568 Bytes vom System belegt
315392 bytes availabe on disk	315392 Bytes freie Kapazität

Wurde der Zusatz /M mit eingegeben:

179712 bytes total disk space	179712 Bytes Gesamtkapazität
179712 bytes availabe on disk	179712 Bytes freie Kapazität

Sämtliche Statusmeldungen haben diese Form. Es ist möglich, daß die Kapazität je nach Diskette variiert.

Nach der Statusmeldung erscheint die Frage:

Do you wish to format another disk (Y/N)?	Wollen Sie eine weitere Diskette formatieren (J/N)?

Geben Sie ein **Y** für Ja ein, beginnt der Vorgang erneut, wie bereits beschrieben. Geben Sie ein **N**(ein) ein, sind Sie wieder im Betriebssystem.

Weitere Erklärungen

Diese Informationen über das Dienstprogramm sind nicht erforderlich, um das Programm aufzurufen oder es ablaufen zu lassen. Diese Informationen werden für den interessierten Leser weitergegeben, der mehr über die Arbeitsweise wissen möchte.

Das Dienstprogramm FORMAT schreibt bereits verschiedene *Informationen* auf die Platte, auch wenn Sie keine Zusatzeingaben machen. Diese Informationen werden benötigt, damit das Betriebssystem Daten auf der Platte speichern und schnell wieder darauf zugreifen kann. Jede Platte enthält in den ersten Sektoren allgemeine Informationen zur Plattenverwaltung. Die Bilder 11.3 bis 11.8 zeigen die Einträge für verschiedene Formate.

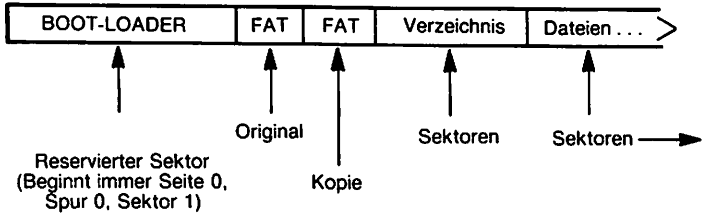


Bild 11.3
Allgemeine Struktur einer formatierten Platte

Außer beim Formatieren einer Winchester, initialisiert FORMAT jeden Sektor der Platte (gilt nicht bei Eingabe von /C). Danach wird immer in den ersten Sektor der Seite 0, Spur 0 der MS-DOS BOOT-LOADER geschrieben. Danach wird im FAT ein erster Eintrag vorgenommen. Dieser hexadezimale Wert kennzeichnet das Format der Platte. Tabelle 11.4 zeigt die möglichen Formate.

Tabelle 11.4 FAT-Eintrag zur Plattenkennzeichnung

FAT-WERT	PLATTENFORMAT
FFh	48-tpi doppelseitig, 8 Sektoren je Spur
FEh	48-tpi einseitig, 8 Sektoren je Spur
FDh	48-tpi doppelseitig, 9 Sektoren je Spur
FCh	48-tpi einseitig, 9 Sektoren je Spur
F9h	96-tpi doppelseitig, 15 Sektoren pro Spur
F8h	Winchesterlaufwerk

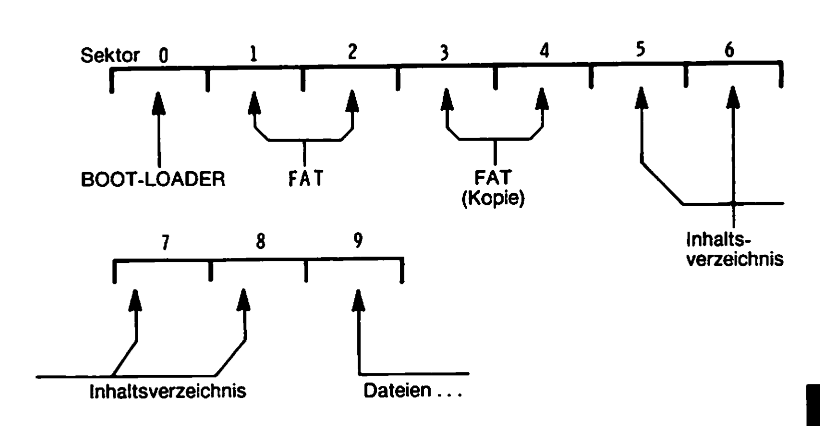


Bild 11.4 Format: SS, 9 Bereiche je Spur

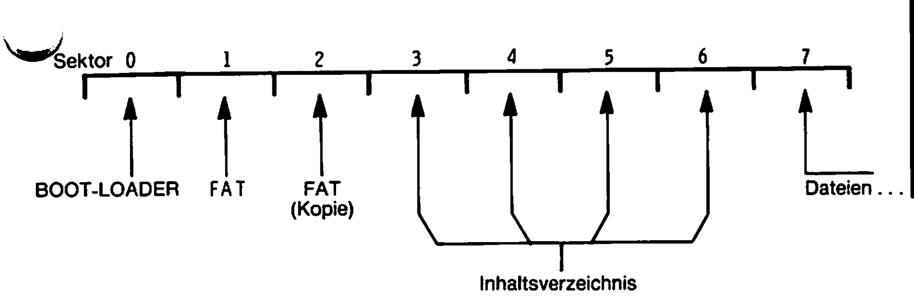


Bild 11.5 Format: SS, 8 Bereiche je Spur

Befehlserklärungen

FORMAT

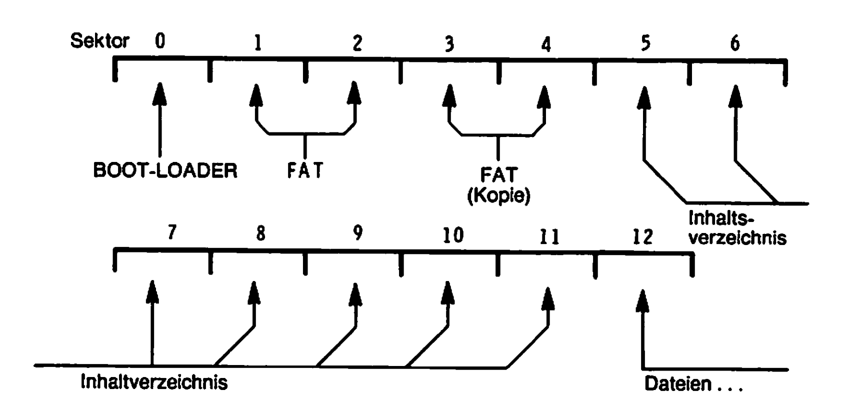


Bild 11.6 Format: DS, 9 Bereiche je Spur

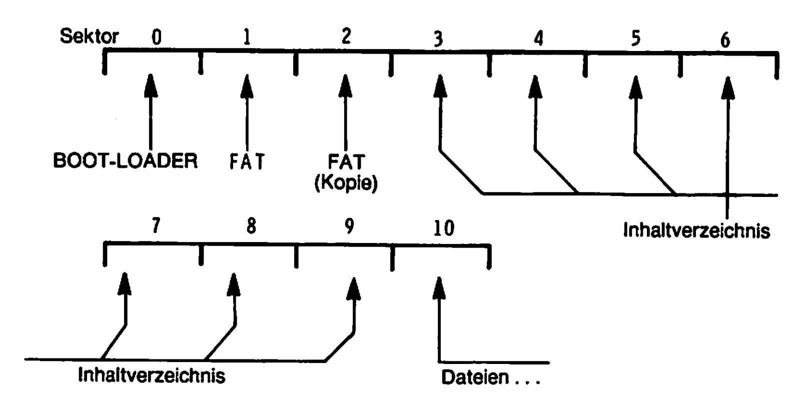


Bild 11.7 Format: DS, 8 Bereiche je Spur

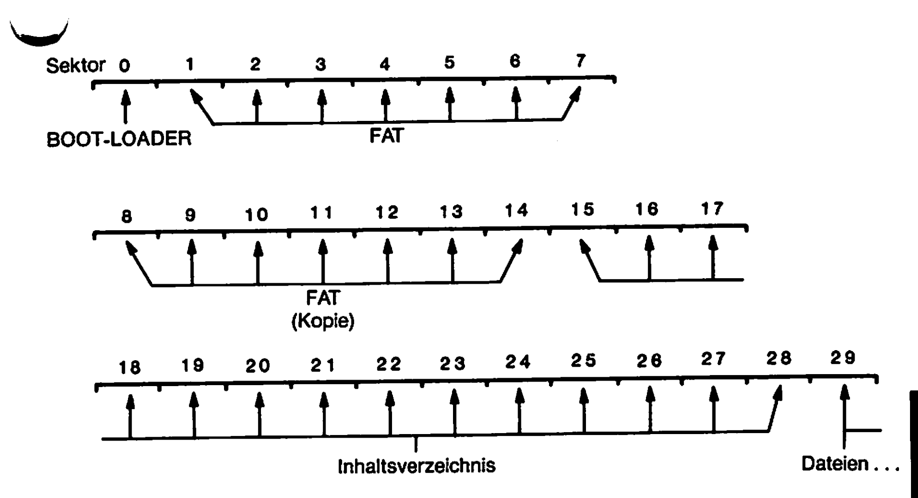


Bild 11.8
Format: DS, 15 Bereiche je Spur

Dateizuordnungstabelle

Die Daten einer Datei werden in *Gruppen* (Cluster) auf der Platte gespeichert. Damit wird eine optimale Nutzung des freien Platzes bei gleichzeitig minimalen Bewegungen des Schreib-Lese-Kopfes erreicht. Der Schreib-Lese-Kopf erfaßt erst alle Daten einer Spur (Zylinder bei der Winchester) bevor er zur nächsten weitergeht.

Der erste Eintrag in der Tabelle kennzeichnet die Art der Platte (s. a. Tabelle 11.4). Das zweite und dritte Byte sind immer FFh. Ab dem dritten Eintrag (Cluster 002) werden die Gruppen der Daten angezeigt. Dabei werden die Daten einer Datei nicht immer hintereinander auf die Platte geschrieben. Wird eine Datei gespeichert, sucht das Betriebssystem immer den nächsten freien Eintrag und speichert dort die Daten. Ist vorher eine Datei gelöscht worden, wird dieser Platz als erstes wieder belegt, unabhängig davon, ob er die gesamten Daten einer Datei speichern kann.

Inhaltsverzeichnis

Vom Programm FORMAT wird ebenfalls das Rootdirectory einer jeden Platte angelegt. Nur dieses Directory kann eine begrenzte Zahl von Dateien speichern. Alle anderen Subdirectories sind Dateien und können eine unbegrenzte Zahl von Einträgen speichern. Jeder Eintrag im Verzeichnis ist 32 Bytes lang und speichert folgende Informationen (Zahl der Bytes in dezimal):

0 - 7 Dateiname; das erste Byte zeigt die Art des Eintrags.

00h wird nicht benutzt. Markiert das Ende des Verzeichnisses.

E5h Dieser Eintrag wurde gelöscht.

2Eh Dieser Eintrag ist das momentane Verzeichnis. Ist das zweite Byte ebenfalls 2Eh, enthält das Gruppenfeld die Gruppennummer des vorherigen Verzeichnisses. (Ist der Wert 0000h, ist dies das Rootdir.)

Jedes andere Zeichen ist ein Zeichen eines Dateinamens.

8 - 10 Dateinamenzusatz

11 Dateiattribut; es ist folgendermaßen definiert:

01 Nur-Lese-Datei. Wird versucht, diese Datei mit der Steuerfunktion 4Dh zu öffnen, wird eine Fehlermeldung ausgegeben. Dieser Wert darf mit anderen zusammen verwendet werden.

02 verdeckte Datei. Sie wird nicht beachtet beim Durchsuchen des Verzeichnisses.

04 System-Datei. Sie wird nicht beachtet beim Durchsuchen des Verzeichnisses.

08 Plattenname. Er enthält außer Tag und Zeit der Anlage keine weiteren Informationen.

10 Subdirectory. Es wird nicht beachtet beim Durchsuchen des Verzeichnisses.

20 Sicherungsbyte. Es wird gesetzt wenn die Datei nach dem Schreiben geschlossen wird. Dieser Wert darf mit anderen zusammen verwendet werden. Er wird von den Programmen BACKUP und RESTORE abgefragt.

Die Dateien IO.SYS und MSDOS.SYS sind als Nur-Lese-Dateien, verdeckte Dateien und System-Dateien gekennzeichnet. Eine Datei kann bei der Anlage als verdeckte Datei markiert werden.

12 - 21 reserviert für DOS

22 - 23 Zeit der Dateianlage oder letzten Überarbeitung

24 - 25 Datum der Dateianlage oder letzten Überarbeitung

26 - 27 relative Gruppennummer der ersten Gruppe der Datei. Der Wert beträgt für die erste Datei der Platte immer 0200h. Der Eintrag wird mit dem niedrigwertigen Byte zuerst gespeichert.

28- 31 Dateigröße in Bytes; das erste Wort enthält den niedrigwertigen Teil der Größe. Beide Wörter werden mit dem niedrigwertigen Byte zuerst gespeichert.

Fehlermeldungen

Cannot FORMAT a Network drive	Ein Laufwerk des Netzwerkes kann nicht formatiert werden

ERKLÄRUNG:
Sie haben versucht, mit dem Befehl FORMAT ein Laufwerk des Netzwerkes zu formatieren. Dies ist nicht möglich.

Disk unsuitable for system disk	Diskette nicht als Systemdiskette zu benutzen

ERKLÄRUNG:
Die Diskette ist fehlerhaft und nicht als Systemdiskette zu benutzen. Die ersten Spuren können die Systemdateien nicht speichern.

Format failure	Formatfehler

ERKLÄRUNG:
Die Diskette ist beschädigt und kann nicht formatiert werden. Eventuell ist der Befehl falsch eingegeben worden.

Insert DOS Disk	Legen Sie Diskette mit Betriebssystem ein

ERKLÄRUNG:
Die Diskette des Arbeitslaufwerkes enthält keine Systemdateien, die kopiert werden können. Legen Sie eine Systemdiskette ein.

Insufficient memory for system transfer	Der Arbeitsspeicher ist für die Systemübertragung zu klein

ERKLÄRUNG:
Der Arbeitsspeicher ist zu klein oder der Systembereich arbeitet nicht einwandfrei. Starten Sie den Befehl neu, oder vergrößern Sie den Arbeitsspeicher.

Invalid characters in volume ID	Falsche Zeichen im Plattennamen

ERKLÄRUNG:
Der Plattenname wurd falsch eingegeben. Überprüfen Sie Ihre Eingabe (s. a. Kapitel 1).

Invalid drive specification	Falsche Laufwerkbezeichnung

ERKLÄRUNG:
Sie haben eine ungültige Laufwerkbezeichnung eingegeben. Wiederholen Sie Ihre Eingabe.

Invalid parameter	Falsche Eingabe

ERKLÄRUNG:
Die Befehlszeile enthält ungültige Zeichen oder Zeichen in falscher Reihenfolge. Überprüfen Sie Ihre Eingabe.

Track 0 bad - disk unusable	Spur 0 beschädigt - Diskette ist nicht zu gebrauchen

ERKLÄRUNG:
Die Spur Null der Diskette ist beschädigt. Es ist nicht möglich, die Diskette zu benutzen.

Cannot format an assigned drive	Ein umgeleitetes Laufwerk kann nicht formatiert werden

ERKLÄRUNG:
Sie haben mit dem Dienstprogramm ASSIGN Laufwerkanfragen für ein oder mehrere Laufwerke umgeleitet. Machen Sie diese Änderungen wieder ungültig. Rufen Sie FORMAT dann erneut auf.

GOTO Intern

Zweck

führt einen unbedingten Sprungbefehl in einer Batchdatei aus

Syntax

GOTO Bezeichnung

Bemerkungen

Dieser Befehl wird in einer Batchdatei abgearbeitet. Nähere Einzelheiten über Batchdateien finden Sie im Kapitel 5.

Findet der Befehlsinterpreter in einer Batchdatei einen GOTO-Befehl, so *springt* er sofort in die dort bezeichnete Zeile. Die Zielzeile des GOTO-Befehles wird durch eine *Zeichenkette* bestimmt, die mit einem Doppelpunkt (:) anfangen muß. Die folgenden acht Zeichen bestimmen dann die Bezeichnung des GOTO-Befehles. Die Programmabarbeitung wird ab der nächsten Zeile nach der Bezeichnung fortgesetzt. Dabei darf die Sprungzeile irgendwo in der Batchdatei stehen. Die Bezeichnung wird während der Abarbeitung der Datei nicht angezeigt. Ein Beispiel:

```
IF EXIST %1 GOTO X
GOTO Y
:X
ECHO Die Datei %1 ist vorhanden.
GOTO ENDE
:Y
ECHO Die Datei %1 ist nicht vorhanden.
:ENDE
```

In der ersten Zeile wird nur bei Erfüllung der Bedingung zur Bezeichnung

:X

gesprungen, d. h. wenn die, mit Aufruf der Batchdatei eingegebene Datei vorhanden ist. Ist die Datei nicht vorhanden, wird immer ab der zweiten Zeile zur Bezeichnung

:Y

gesprungen. Ist die Datei vorhanden, wird dies durch die Zeile

ECHO Die Datei ist vorhanden.

angezeigt. Danach wird immer zur Bezeichnung ENDE gesprungen.

Ist die Bezeichnung in der Datei nicht vorhanden, erscheint eine Fehlermeldung und die Datei wird sofort verlassen.

Sie können auch Zeilen, die mit einem Doppelpunkt beginnen, als Kommentarzeilen in die Batchdatei einfügen. Die Kommentare werden bei der Abarbeitung nicht beachtet und auch nicht auf dem Bildschirm angezeigt.

Fehlermeldung

Label not found	Bezeichnung nicht vorhanden

ERKLÄRUNG:
Die hinter dem GOTO-Befehl genannte Bezeichnung ist in der Datei nicht vorhanden.

GRAFTABL Extern

Zweck

lädt die ASCII-Zeichen mit den Werten 128 - 255 für den Graphikmodus

Syntax

[d:] [Weg] GRAFTABL

dabei bedeutet:

d: vor dem Programmnamen bestimmt das aufzurufende Laufwerk

Weg vor dem Programmnamen bestimmt das aufzurufende Verzeichnis

Bemerkungen

Das Programm GRAFTABL kann im *Netzwerk* **ohne** Einschränkungen benutzt werden.

Ihr Zenith PC kann maximal 255 verschiedene Zeichen darstellen. Für den Textmodus des Graphik- oder Monochromeadapters sind diese Zeichen komplett in einem ROM gespeichert. Für den Graphikmodus sind dort nur die ersten 128 Zeichen gespeichert. Damit haben Sie die Möglichkeit, selber eigene Zeichen zu definieren. Weitere Einzelheiten dazu erfahren Sie im Programmierhandbuch der Version 3.

Das Programm GRAFTABL hat die Zeichendefinitionen für die ASCII-Zeichen 128 bis 255 gespeichert. Rufen Sie das Programm auf mit der Eingabe von:

GRAFTABL

Das Programm nimmt alle notwendigen Änderungen im Systemspeicher vor und verbleibt dann als Erweiterung des Betriebssystemes im Speicher. Es belegt ca. 1 KByte.

Haben Sie mit dem Befehl SELECT das Programm KEYBGR ausgewählt, sollten Sie in der Datei AUTOEXEC.BAT zusätzlich das Programm GRAFTABL aufrufen. Nur dann werden auch im Graphikmodus deutsche Sonderzeichen dargestellt.

Wird das Programm zum ersten Mal geladen, erscheint die Meldung:

Graphics characters loaded	Definition für Graphikzeichen geladen

Rufen Sie das Programm danach erneut auf, kann es nicht mehr geladen werden. Es erscheint die Fehlermeldung:

Graphics characters already loaded	Definition für Graphikzeichen bereits geladen

IF **Intern**

Zweck

macht die Befehlsausführung von der Erfüllung einer Bedingung abhängig

Syntax

IF NOT Bedingung Befehl

dabei bedeutet:

NOT verneint die Bedingung

Bedingung eine Eingabe aus der folgenden Liste:

ERRORLEVEL *Nummer*, Ausgangskode des Programmes, daß gerade verlassen wurde

Zeichenkette1==Zeichenkette2, die Bedingung wird nur erfüllt, wenn beide Zeichenketten gleich sind

EXIST *Dateibez*, die Bedingung wird nur erfüllt, wenn die Datei vorhanden ist

Befehl ein auszuführender DOS-Befehl

Bemerkungen

Mit dem Befehl IF wird die Ausführung eines Befehls in einer Batchdatei von der *Erfüllung* einer Bedingung abhängig gemacht. Wird die Bedingung erfüllt, und nur dann, wird der nachfolgende Befehl ausgeführt. Sonst wird er nicht beachtet. Durch die Eingabe von NOT kann das Gegenteil erreicht werden. Nur wenn die Bedingung nicht erfüllt wird, ,und nur dann, wird der nachfolgende Befehl ausgeführt.

Dieser nachfolgende Befehl kann ein GOTO-Befehl sein, der einen Sprung innerhalb der Datei veranlaßt (s. a. Seite 11.166).

Damit eine einmal geschriebene Batchdatei flexibel eingesetzt werden kann, ist es möglich, Variable zu verarbeiten. Es kann eine beliebige Anzahl von Variablen eingesetzt werden (unter MS-DOS 1.25 sind nur 10 Variable möglich). Die Variablen werden in der Datei mit den Ziffern 0 bis 9 (gleich 10 Variable) und einem vorangestellten Prozentzeichen (%) bezeichnet. Die entsprechenden Werte dafür werden in der Befehlszeile hinter dem Dateinamen eingegeben. Jeder Wert wird durch ein Freizeichen getrennt. Dabei werden die eingegebenen Werte in ihrer Reihenfolge in der Datei ersetzt. Der erste zu ersetzende Wert ist der Dateiname selber (%0). Dadurch ist es möglich, daß sich die Datei selber wieder aufruft. Ein Beispiel:

```
Dateiname   Variable 1   Variable 2   Variable 3 ...   Variable 9
!           !            !            !                !
!           !            !            !                !
%0          %1           %2           %3 ...           %9
```

Mit dem Befehl SHIFT ist es möglich, diese Begrenzung aufzuheben. Jeder Aufruf von SHIFT in der Datei verschiebt die Variablen um eine Stelle nach links. Damit wird aus der 11. Variablen die 10. oder aus der 3. Variablen die 2 (s. a. Seite 11.279).

Der Befehl IF kann drei verschiedene Arten von Bedingungen abfragen:

1. ERRORLEVEL Nummer

Die Bedingung wird erfüllt, wenn das Programm mit einem Ausgangskode verlassen wird, dessen Wert der *Nummer* entspricht, oder der höher ist.

2. Zeichenkette1 == Zeichenkette2

Die Bedingung wird erfüllt, wenn der Inhalt beider Zeichenketten gleich ist. Wird für eine Zeichenkette eine Variable eingesetzt, wird erst die Variable ersetzt und dann die Bedingung überprüft.

3. EXIST *Dateiname*

Die Bedingung wird erfüllt, wenn die bezeichnete *Datei* auf der Platte vorhanden ist.

Achten Sie darauf, daß Sie bei allen Eingaben die Befehlsworte immer mit *Großbuchstaben* eingeben. Nur so kann der Befehl richtig abgearbeitet werden, andernfalls erscheint die Fehlermeldung:

```
Syntax error                    Schreibfehler
```

Beispiele

Die folgenden Abschnitte zeigen einige mögliche Beispiele für Batchdateien. Zusätzliche Informationen finden Sie im Kapitel 5.

Beispiel 1

Es ist folgende Batchatei mit dem Namen CHECK.BAT vorhanden:

ECHO OFF
CHKDSK %1:
IF ERRORLEVEL 0 ECHO Der Ausgangskode des Programmes ist 0.

Rufen Sie die Datei zusammen mit dem Buchstaben für das zu überprüfende Laufwerk auf. Nach der Statusmeldung des Programmes CHKDSK erscheint folgender Text:

```
Der Ausgangskode des Programmes ist 0.
```

Beispiel 2

Die folgende Batchdatei hat den Namen STREICHE.BAT:

ECHO OFF
:ANFANG
IF "%1" == "" GOTO ENDE
DEL %1
ECHO Die Datei %1 ist gelöscht.
SHIFT
GOTO ANFANG
:ENDE

Mit dieser Batchdatei werden sämtliche in der Befehlszeile eingegebenen Dateien gelöscht. In Zeile drei der Datei wird erst die Variable %1 durch einen Dateinamen ersetzt und dann die Bedingung abgefragt. Kann für die Variable ein Dateiname eingesetzt werden, wird die Bedingung nicht erfüllt. Die Datei wird mit der nächsten Zeile (DEL %1) fortgesetzt. Ist keine Datei mehr vorhanden, ist der Wert der Variablen (%1) gleich " ". Damit wird die Bedingung erfüllt, und der zweite Teil des Befehls (GOTO ENDE) wird ausgeführt.

Beispiel 3

Die folgende Batchdatei mit dem Namen ZEIGE.BAT überprüft das Inhaltsverzeichnis der Platte auf das Vorhandensein der eingegebenen Datei. Die Datei hat den Inhalt:

```
ECHO OFF
IF EXIST %1 GOTO X
ECHO Die Datei ist nicht vorhanden
GOTO END
:X
ECHO Die Datei ist vorhanden
:END
```

Fehlermeldungen

Invalid number of parameters — Falsche Anzahl der Zusatzeingaben

ERKLÄRUNG:
Sie haben nicht alle Teile des Befehls IF eingegeben oder versucht, mehr als einen FOR-Befehl in einer Zeile einzugeben. Überprüfen Sie Ihre Eingabe.

Syntax Error — Schreibfehler

ERKLÄRUNG:
Sie haben den Befehl IF nicht richtig geschrieben. Überprüfen Sie Ihre Eingabe.

JOIN Extern

Zweck

ersetzt eine Laufwerkbezeichnung vorübergehend durch einen Verzeichnisnamen

Syntax

[d:] [Weg] JOIN
[d:] [Weg] JOIN d: [d:] Weg
[d:] [Weg] JOIN d: /D

dabei bedeutet:

d: vor dem Programmnamen bestimmt das aufzurufende Laufwerk

Weg vor dem Programmnamen bestimmt das aufzurufende Verzeichnis

d: nach dem Programmnamen bestimmt das zu ersetzende Laufwerk

Weg nach dem Programmnamen bestimmt das neue Verzeichnis in der Form:

[\] [Verzeichnis [\Verzeichnis...]]

/D löscht eine Umbenennung

Bemerkungen

Das Programm JOIN kann im *Netzwerk* **nicht** benutzt werden.

Mit dem Programm JOIN können Sie ein komplettes Laufwerk mit allen Unterverzeichnissen durch einen Verzeichnisnamen ersetzen. Dies kann erforderlich werden, haben Sie sich ein tief gestaffeltes System von Unterverzeichnissen geschaffen. Viele Anwendungsprogramme erlauben nur eine bestimmte Zeichenzahl (ca. 30) für den kompletten Dateinamen (Laufwerk, Verzeichnis, Dateiname). Schaffen Sie mit dem Programm JOIN ein neues Verzeichnis, können die Programme wieder problemlos arbeiten.

ACHTUNG:
Sie können dieses Programm nicht aufrufen, wenn Sie mit dem Dienstprogramm ASSIGN Anfragen für das Laufwerk vorübergehend umgeleitet haben. Ebenso sollten Sie die Programme BACKUP, DISKCOPY, DISKCOMP und RESTORE nicht für Laufwerke und Verzeichnisse benutzen, die mit JOIN bearbeitet wurden.

Damit das Programm JOIN ein Laufwerk komplett durch ein neues Verzeichnis ersetzen kann, müssen folgende Voraussetzungen eingehalten werden:

- Das Arbeitslaufwerk kann nicht ersetzt werden.
- Das Arbeitsverzeichnis kann nicht als neues Verzeichnis benannt werden.
- Das neue Verzeichnis muß im Root-Dir angelegt werden.
- Im neuen Verzeichnis dürfen keine Dateien gespeichert sein.

Da das Programm JOIN den bisherigen Laufwerknamen komplett ersetzt, kann das Laufwerk anschließend mit seinem ursprünglichen Namen nicht mehr aufgerufen werden. Dies gilt, bis die Umbenennung gelöscht wird, oder Sie das System neu starten.

JOIN

Programmaufruf

Es sind drei verschiedene Befehlseingaben möglich.

1. Rufen Sie das Programm nur mit seinem Namen auf, erhalten Sie eine Übersicht der momentanen Umbenennungen. Geben Sie ein:

   ```
   JOIN
   ```

   Es könnte z.B. folgende Übersicht erscheinen:

   ```
   A: => C:\WORDS
   B: => C:\NUMBERS
   ```

2. Ersetzen Sie ein Laufwerk durch einen Verzeichnisnamen. Dabei beachten Sie, daß weder das aktuelle Laufwerk ersetzt werden kann, noch daß das aktuelle Verzeichnis der neue Name für das Laufwerk wird. Geben Sie ein:

   ```
   JOIN A: C:\TEXT
   ```

   Jetzt wird das Laufwerk A: durch das Verzeichnis \TEXT auf dem Laufwerk C: ersetzt. Damit kann das Laufwerk A: nicht mehr direkt benutzt werden.

3. Löschen Sie eine Umbenennung wieder. Rufen Sie den Befehl mit dem ursprünglichen Laufwerknamen und dem Zusatz /D auf. Geben Sie ein:

   ```
   JOIN A: /D
   ```

   Jetzt wird die Umbenennung gelöscht, und Sie können das Laufwerk A: mit seinem richtigen Namen erreichen.

Fehlermeldungen

Cannot JOIN a Network drive	Ein Laufwerk des Netzwerkes kann nicht ersetzt werden

ERKLÄRUNG:
Sie haben versucht mit dem Befehl JOIN ein Laufwerk des Netzwerkes durch einen Verzeichnisnamen zu ersetzen. Dies ist nicht möglich.

Directory not empty	Verzeichnis nicht leer

ERKLÄRUNG:
Das neu benannte Verzeichnis ist nicht leer. Löschen Sie die dort gespeicherten Dateien, oder benennen Sie ein anderes Verzeichnis. Benutzen Sie trotzdem dieses Verzeichnis, können Sie die dort gespeicherten Dateien vorübergehend nicht erreichen.

Incorrect number of parameters	Falsche Anzahl der Zusatzeingaben

ERKLÄRUNG:
Sie haben den Befehl mit falschen Zusatzeingaben aufgerufen. Im Abschnitt Programmaufruf finden Sie eine Übersicht der drei möglichen Eingabezeilen.

Not enough memory	Arbeitsspeicher zu klein

ERKLÄRUNG:
Das Programm JOIN erweitert das Betriebssystem, damit es eine Tabelle mit den Umbenennungen speichern kann. Der freie Arbeitsspeicher ist für diese Tabelle nicht mehr groß genug. Löschen Sie resident gespeicherte Programme, oder reservieren Sie mit dem Befehl BUFFERS weniger Zwischenspeicher.

KEYBxxxx Extern

Zweck

lädt ein Programm zur Tastaturanpassung

Syntax

[d:] [Weg] KEYBxxxx

dabei bedeutet:

d: vor dem Programmnamen bestimmt das aufzurufende Laufwerk

Weg vor dem Programmnamen bestimmt das aufzurufende Verzeichnis

xxxx Landesbezeichnung für das zu ladende Programm (s. a. Tabelle 11.4)

Bemerkungen

Im *ROM* Ihres Zenith PC ist eine Routine enthalten, die jeden Tastendruck der Tastatur untersucht und anschließend einen entsprechenden Wert erzeugt (s. a. Bedienerhandbuch). Dabei werden auch Tastaturstatus und Sondertasten berücksichtigt. Diese Routine ist für eine amerikanische Tastatur programmiert worden. Für eine Tastatur nach deutscher DIN-Norm wird ein anderes Programm benötigt.

Das Programm *KEYBGR* (s. a. Tabelle 11.4) ersetzt nach seinem Aufruf die Routine im ROM und erzeugt eine deutsche Tastatur. Damit wird der Systemspeicher um ca. 2 KBytes vergrößert. Sie können dieses oder jedes andere Programm der Tabelle 11.4 jederzeit aufrufen. Es wird sofort berücksichtigt (kein erneuter Systemstart erforderlich).

Mit der Tastenkombination **CTRL-ALT-F1** können Sie jede landesspezifische Tastaturanpassung wieder rückgängig machen. Sie haben dann eine US-ASCII-Tastatur. Mit der Tastenkombination **CTRL-ALT-F2** können Sie die geladene Tastaturanpassung (z. B. KEYBGR) wieder aufrufen. Sie haben dann eine landesspezifische Tastatur.

Zeichensatzauswahl

Auf der zweiten Systemdiskette finden Sie mehrere verschiedene Anpassungsprogramme. Dem ersten Teil des Programmnamens KEYB können Sie eine Abkürzung aus maximal weiteren vier Buchstaben anhängen und bestimmen damit das Programm. Es stehen für jedes Land zwei Zeichensätze zur Verfügung.

Von fast allen kommerziellen Programmen wird der erweiterte ASCII-Satz mit 256 Zeichen gefordert. Er enthält verschiedene Graphikzeichen und auch deutsche Sonderzeichen, wie ä oder ö. Dies entspricht dem Zeichensatz des IBM-PC. Den deutschen Zeichensatz enthält das Programm KEYBGR.

Tabelle 11.4 Tastaturprogramme

LAND	7-Bit	8-Bit
Australien	AUS	US
Dänemark	ADA	DA
Deutschland	AGR	GR
Finnland	ASW	SW
Frankreich	AFR	FR
Großbritanien, UK	AUK	UK
Großbritanien, US	AUS	US
Italien	AIT	IT
Kanada, EL	---	CANE
Kanada, SE	---	CANS
Niederlande	AUS	US
Norwegen	ANO	NO
Schweden	ASW	SW
Schweiz, DT	---	CHG
Schweiz, FR	---	CHF
Spanien	ASP	SP

Für einige spezielle Programme oder Drucker ist jedoch der original ASCII-Satz mit 128 Zeichen erforderlich. Dieser ist im Programm KEYBAGR enthalten. Auch dort finden Sie deutsche Sonderzeichen, aber keine Graphikzeichen.

Fehlermeldungen

Internal table size do not match	Interne Tabelle ist fehlerhaft

ERKLÄRUNG:
Die interne Tabelle zur Tastaturanpassung ist fehlerhaft. Das Programm kann nicht arbeiten. Verwenden Sie eine Kopie.

Unable to load Key map routine	Programm kann nicht geladen werden

ERKLKÄRUNG:
Die Routine des ROM wurde bereits durch ein anderes Programm zur Zeichensatzanpassung ersetzt. Starten Sie das System ohne diese Routine neu, und rufen Sie dann das gewünschte Programm auf.

LABEL Extern

Zweck

speichert, ändert oder löscht einen Diskettennamen

Syntax

[d:] [Weg] LABEL [d:] [Name]

dabei bedeutet:

d:	vor dem Programmnamen bestimmt das aufzurufende Laufwerk
Weg	vor dem Programmnamen bestimmt das aufzurufende Verzeichnis
d:	nach dem Programmnamen bestimmt das zu bearbeitende Laufwerk
Name	Plattenname, der gespeichert werden soll

Bemerkungen

Das Programm LABEL kann im *Netzwerk* **nicht** benutzt werden.

Der interne Befehl DIR zeigt nicht nur Dateieinträge des gewünschten Inhaltsverzeichnisses, sondern auch zusätzlich immer in der ersten Zeile den Namen der Platte (sofern gespeichert). Es erscheint folgende Zeile:

Volume in drive x has no label

oder

Volume in drive x is ...

LABEL

Das Dienstprogramm FORMAT fragt nach erfolgreichem Formatieren der Platte immer nach einem Namen, der aus maximal 11 Zeichen bestehen darf.. Sie können diese Anfrage auch übergehen. Geben Sie hier einen Namen ein, wird er auf der Platte gespeichert und mit dem Befehl DIR automatisch gezeigt.

Das Dienstprogramm LABEL ermöglicht es Ihnen, diesen Namen zu ändern oder zu löschen. Ebenfalls können Sie einen neuen Namen vergeben, sofern noch keiner gespeichert ist.

Programmaufruf

Es sind zwei verschiedene Befehlseingaben möglich.

1. Rufen Sie das Programm nur mit seinem Namen auf, zeigt es Ihnen den momentanen Namen des aktuellen Laufwerkes, sofern vorhanden. Geben Sie ein:

   ```
   LABEL
   ```

   Es könnte z. B. der folgende Name gezeigt werden:

   ```
   Volume in drive C: is ZENITH -- 1
   ```

   Darunter erscheint die Aufforderung, einen neuen Namen einzugeben:

   ```
   Enter desired volume label (11 characters, RETURN for none)?
   ```

   Für den Plattennamen gelten diesselben Bestimmungen wie für Dateinamen. Geben Sie einen Namen mit mehr als 8 Zeichen immer ohne Punkt ein, da ein Namenszusatz nicht existiert. Betätigen Sie nur die Taste RETURN, erscheint folgende Frage:

   ```
   Delete current volume label (Y/N)?
   ```

   Sie können jetzt einen vorhandenen Namen löschen, ohne gleichzeitig einen neuen einzugeben. Betätigen Sie dafür die Taste Y. Bei Eingabe eines N für Nein, bleibt der alte Name unverändert bestehen.

Kann der neue Plattenname fehlerfrei gespeichert werden, erscheint keine weitere Meldung, und Sie sind wieder im Betriebssystem.

2. Rufen Sie das Programm mit einem anderen Zusatz als einer Laufwerkbezeichnung auf, ist dies ein neuer Plattenname. Das Programm LABEL speichert dann diese Eingabe auf dem aktuellen oder bezeichneten Laufwerk als neuen Namen. Geben Sie z. B. ein:

```
LABEL D:--TEST--
```

Der neue Name des Laufwerkes D: ist --TEST--. Bei dieser Eingabe kann das aktuelle Laufwerk jedes beliebige andere Laufwerk sein. Alle Zeichen nach dem Doppelpunkt der Laufwerkbezeichnung bis zum ersten Leerfeld werden als neuer Plattenname genommen. Enthält die Befehlszeile nur gültige Zeichen, wird der neue Name gespeichert. Bei falschen Zeichen erscheint eine Fehlermeldung mit der Aufforderung, den Namen korrekt einzugeben.

Fehlermeldungen

Cannot LABEL a Network drive	Einem Laufwerk des Netzwerkes kann kein Name gegeben werden

ERKLÄRUNG:
Sie haben versucht, mit dem Befehl LABEL einem Laufwerk des Netzwerkes einen neuen Plattennamen zu geben. Dies ist nicht möglich.

Invalid characters in volume label	Falsche Zeichen im Plattennnamen

ERKLÄRUNG:
Der eingegebene Name enthält ungültige Zeichen, z. B. die Zeichen \, > oder <. Auch ein Punkt ist nicht erlaubt. Geben Sie den Namen neu ein.

LABEL

Cannot LABEL a SUBSTed or ASSIGNed drive — Einem umgeleiteten Laufwerk kann kein Name gegeben werden

ERKLÄRUNG:
Sie haben versucht, mit dem Befehl LABEL einem Laufwerk, das mit den Befehlen ASSIGN oder SUBST umgeleitet wurde, einen neuen Plattennamen zu geben. Dies ist nicht möglich.

Invalid drive specification — Falsche Laufwerkbezeichnung

ERKLÄRUNG:
Sie haben den Befehl LABEL für ein nicht vorhandenes Laufwerk aufgerufen. Geben Sie in der Befehlszeile nur gültige Laufwerknamen ein.

No room in root directory — Kein Platz im Root-Dir

ERKLÄRUNG:
Das Programm LABEL benötigt zur Speicherung des Namens einen freien Eintrag im Root-Dir der gewünschten Platte. Das Verzeichnis dieser Platte ist bereits voll. Löschen Sie mindestens einen Eintrag, rufen Sie das Programm dann neu auf.

Too many files open — Keine Dateinummer mehr frei

ERKLÄRUNG:
Das Programm LABEL benutzt bei der Speicherung des Plattennamens von MS-DOS vergebene Dateinummern. Die maximal verfügbare Zahl ist bereits vergeben. Sie müssen in der Datei CONFIG.SYS mit dem Befehl FILES eine größere Zahl von Dateinummern bestimmen.

Incorrect DOS version — Falsche DOS-Version

ERKLÄRUNG:
Sie benutzen das Programm LABEL nicht unter einer DOS-Nummer die 3.0 oder höher ist.

MKDIR oder MD — Intern

Zweck

legt ein neues Inhaltsverzeichnis an

Syntax

MKDIR D: Weg
MD D: Weg

dabei bedeutet:

D: Bezeichnung des Laufwerks, auf dem das neue Verzeichnis angelegt werden soll.

Weg mehrere Inhaltsverzeichnisse in Folge in der Form:

[\] [Verzeichnis [\Verzeichnis ...]]

Bemerkungen

Mit diesem Befehl werden neue *Inhaltsverzeichnisse* angelegt. Diese Verzeichnisse werden dann *Subverzeichnisse* (Subdirectory) genannt. Sie haben damit die Möglichkeit, ein strukturiertes Inhaltsverzeichnis der Platte zu erzeugen. Nähere Erläuterungen dazu im Kapitel 7.

Der Befehl kann mit seinem vollen Namen oder auch mit der Abkürzung eingegeben werden. Geben Sie folgenden Befehl ein:

MD C:\TEXT

Damit wird im Rootdir des Laufwerkes C: das Subdir TEXT angelegt. In diesem Subdir werden sofort von MS-DOS zwei Einträge vorgenommen. Dies sind die Einträge:

```
.     <DIR> 08-01-84   12:04p
..    <DIR> 08-01-84   12:04p
```

MKDIR oder MD

Im ersten Eintrag (ein Punkt) speichert DOS den Namen des Verzeichnisses selber. Im zweiten Eintrag (zwei Punkte) speichert DOS den Namen des Verzeichnisses, von dem aus dieses Subverzeichnis angelegt wurde. DOS benötigt diese beiden Einträge, um den Weg vom Verzeichnis zum Rootdir zu finden. Die Einträge können von Ihnen nicht gelöscht oder umgeschrieben werden. Sie werden erst gelöscht, wenn das Verzeichnis mit dem Befehl RD gelöscht wird.

Mit dem Befehl MD oder MKDIR können Sie auch Verzeichnisse in weiter entfernt liegenden Subverzeichnissen anlegen. Ein Beispiel:

```
MKDIR TEXT\SPEICHER\TEST
```

Sie gehen mit diesem Befehl vom momentanen Subdir über das Subdir TEXT zum Subdir SPEICHER. Dort wird das neue Subdir TEST angelegt. Ein weiteres Beispiel

```
MD SYSTEM.COM
```

legt das Subverzeichnis mit dem Namen SYSTEM.COM an. Im Namen des Subdir dürfen Sie alle gültigen MS-DOS-Zeichen für Dateinamen verwenden (s. a. Kapitel 1).

Fehlermeldungen

Invalid number of parameters — Falsche Anzahl von Zusatzeingaben

ERKLÄRUNG:
Sie haben den Befehl nicht komplett eingegeben. Überprüfen Sie Ihre Eingabe.

Unable to create directory — Das Verzeichnis kann nicht angelegt werden

ERKLÄRUNG:
Sie haben einen Namen eingegeben, der schon für eine Datei oder ein anderes Subverzeichnis in diesem Verzeichnis gespeichert ist. Überprüfen Sie Ihre Eingabe.

MODE Extern

Zweck

bestimmt die Betriebsart für den Bildschirm, monochrome oder graphisch; bestimmt das Protokoll für den parallelen oder seriellen Ausgang. Es sind vier verschiedene Befehlseingaben möglich.

Syntax

MODE ?

dabei bedeutet:

?	zeigt eine ausführliche Erklärung des Programmes

MODE LPT#:[n] [,m] [,P]

dabei bedeutet:

#	1, 2 oder 3, bestimmt den anzupassenden Ausgang LPT1:, LPT2: oder LPT3:
n	80 oder 132, bestimmt die Zahl der Zeichen pro Zeile
m	6 oder 8, bestimmt die Zahl der Zeilen pro Inch
P	bestimmt, daß der Drucker wiederholt angerufen wird, auch wenn er nicht antwortet. Diese Funktion kann nur mit **CTRL-C** beendet werden.

MODE [n] [,[m] [,[T]] [,s]]

dabei bedeutet:

n	bezeichnet den Videomodus
m	R oder L, verschiebt das gesamte Bild nach rechts oder links
T	ruft eine Testzeile zur Bildjustierung auf
s	bestimmt den SCROLL-Modus

MODE

MODE COMn:Baud-Rate [Parity] [,[Wort-Länge] [,Stop-Bit] [,P]]]]

dabei bedeutet:

n	1 oder 2, bestimmt den anzupassenden Seriellausgang
Baud-Rate	110, 150, 300, 600, 1200, 2400, 4800 oder 9600, bestimmt die Übertragungsgeschwindigkeit (Baud-Rate)
Parity	O (Odd), E (Even) oder N (No), bestimmt die Art der Parity
Wort-Länge	7 oder 8, bestimmt die Wortlänge (Data-Bits, errechnet ohne Stop- oder Prüf-Bit)
Stop-Bit	1 oder 2, bestimmt die Anzahl der erforderlichen Stop-Bits
P	bestimmt, daß der Drucker wiederholt angerufen wird, auch wenn er nicht antwortet. Diese Funktion kann nur mit <**CTRL-C**> beendet werden.

MODE LPT#:=COMn

dabei bedeutet:

#	1, 2 oder 3, bestimmt den umzuleitenden Parallelausgang (LPT1:, LPT2: oder LPT3:)
n	1 oder 2, bestimmt den Seriellausgang, zu dem die Daten gesendet werden (COM1: oder COM2:)

Bemerkungen

Das Programm MODE ermöglicht es Ihnen, das Betriebssystem entsprechend den angeschlossenen Peripheriegeräten zu modifizieren. Dies ist ebenfalls mit dem Dienstprogramm CONFIGUR möglich. Das Programm CONFIGUR ist menügesteuert und erlaubt eine Speicherung der vorgenommenen Änderungen auf Diskette. Das Programm MODE benötigt nur einen einzigen Eingabebefehl, speichert aber die Änderungen nicht dauerhaft auf Diskette.

Alle Ein-/Ausgaben werden unter MS-DOS als Informationsaustausch zwichen Peripheriegeräten (Devices) behandelt. Peripheriegeräte sind z. B. die Tastatur, der Bildschirm, Laufwerke, Drucker und Modems.

Der Teil des Betriebssystemes, der für die Steuerung der Peripheriegeräte und den reibungslosen Datenaustausch zwischen ihnen zuständig ist, wird I/O-Handler genannt. Dies ist hauptsächlich die Datei IO.SYS. Nähere Einzelheiten finden Sie in den Kapiteln 8 und 9.

Der I/O-Handler benötigt zum ordnungsgemäßen Arbeiten ein Hilfsprogramm, das auf speziellen Eigenarten des angeschlossenen Gerätes abgestimmt ist. Dieses Hilfsprogramm wird Gerätetreiber (Device-Driver) genannt. Es beinhaltet alle Informationen, die das Betriebssystem über das Peripheriegerät wissen muß. Für die beiden logischen Ausgänge (PRN und AUX) Ihres Zenith PC gibt es je einen Gerätetreiber. Dieser Treiber kann verändert werden (mit den Programmen CONFIGUR und MODE) und steuert mehrere physikalische Ausgänge.

Diese Art der Programmierung erreicht mit wenig Aufwand ein Maximum an Flexibilität. Obwohl an einen logischen Ausgang verschiedene Peripheriegeräte angeschlossen werden können, ist es nicht jedesmal erforderlich, dafür das Betriebssystem neu zu schreiben. Es werden nur die notwendigen Werte im Treiber geändert.

Mit Hilfe des Programmes MODE können Sie schnell und einfach diese Werte im jeweiligen Treiber ändern. Damit werden dann vom Betriebssystem die gerätespezifischen Werte des Peripheriegerätes benutzt. Dies kann z. B. das Übertragungsprotokoll für einen seriellen Drucker sein. Das Protokoll bestimmt die Art des Datenaustausches zwischen Ihrem Zenith PC und dem angechlossenen Gerät.

Programmaufruf

Das Programm MODE kann mit fünf verschiedenen Befehlseingaben aufgerufen werden. Sie können damit folgende Funktionen wählen:

- ausführliche Erklärungen aufrufen
- das Übertragungsprotokoll für einen parallelen Ausgang bestimmen

MODE

- den Bildschirmausgang für eine Grafik- oder Monochromekarte bestimmen
- das Übertragungsprotokoll für einen seriellen Ausgang bestimmen
- einen parallelen Ausgang zu einem seriellen Ausgang umleiten

Erklärungen

Rufen Sie die Programmerläuterungen auf durch Eingabe von

MODE ?

und es erscheint folgendes Bild:

```
                    MODE Version 3.02
        Copyright (C) 1985 Zenith Data Systems Corporation
MODE enables you to configure your system for specific peripheral devices by entering
a single command line. The configuration or protocol that you define with MODE
remains in memory for use by the system until you redefine it with another MODE
command or until you reset or reboot your system.

Syntax: MODE ?
        MODE LPT#:[n][,[m][,P]]
        MODE [n][,[m][,[T]][,s]]
        MODE COMn:baud[,[parity][,[databits][,[stopbits][,P]]]]
        MODE LPT#:=COMn

The command lines shown above are used to

1 -- Display this help screen
2 -- Configure a parallel device
3 -- Configure a color/graphics or monochrome display device
4 -- Configure a serial device
5 -- Remap parallel output to a serial port

For more Information, enter the number of your selection. To exit.
press RETURN.
        Enter selection or press RETURN:
```

Es erscheint eine allgemeine Übersicht des Programmes mit einer Auswahl von weiteren Erläuterungen. Durch Eingabe der gewünschten Ziffer können Sie diese Erklärungen aufrufen, die Ziffer 1 ruft diese allgemeine Übersicht auf, die Ziffern 2 bis 5 rufen weitere Erklärungen zur Befehlseingabe auf. Das Programm verlassen Sie wieder durch Betätigen der Taste **RETURN**.

Geben Sie eine **2** ein, erscheinen Erklärungen zur Anpassung eines Parallelausganges:

MODE Version 3.02
Configuring a Parallel Device

To configure a parallel device, enter a command line in the form

MODE LPT#:[n][,[m][,P]]

where # is 1, 2, or 3 and designates the specific LPT device (LPT1:, LPT2: or LPT3:) to be configured;
n is 80 or 132, and specifies the number of characters per line;
m is 6 or 8, and specifies the number of lines per inch; and
P invokes continuous retry on timeout errors.

Defaults are 80 characters per line and 6 lines per inch.

Press any key to return to the main menu or RETURN to exit:

Betätigen Sie jetzt die Taste RETURN verlassen Sie das Programm wieder, mit Betätigen jeder anderen Taste kehren Sie zur allgemeinen Übersicht zurück.

Durch Eingabe einer **3** können Sie Erklärungen zur Bildschirmanpassung aufrufen. Diese Erklärungen sind sehr umfangreich und erstrecken sich über zwei Seiten. Durch Betätigen einer beliebigen Taste rufen sie die zweite Seite auf. Auch hier können Sie das Programm durch Eingabe von **RETURN** verlassen.

MODE

MODE Version 3.02
Configuring a Display Device

To configure a color/graphics or monochrome display device, enter a command line in the form

MODE [n][,[m][,[T]][,s]]

where n defines the display mode and may be one of the following:

40 Sets the display width to 40 characters per line.
80 Sets the display width to 80 characters per line.
BW40 Sets the display mode to black and white (disables color burst) with 40 characters per line
BW80 Sets the display mode to black and white (disables color burst) with 80 characters per line.
CO40 Sets the display mode to color (enables color burst) with 40 characters per line.
CO80 Sets the display mode to color (enables color burst) with 80 characters per line.
GR40 Sets the display mode to medium-resolution graphics mode with 40 characters per line.
GR80 Sets the display mode to medium-resolution graphics mode with 80 characters per line.
MONO Sets the display mode to monochrome. In this mode, the screen display width is always 80 characters per line.
press any key to continue:

MODE Version 3.02
MODE [n][,[m][,[T]][,s]]

m shifts the display, and may be either R (right) or L (left). For an 80-character display, the m parameter shifts the display two character positions. For a 40-character display, the m parameter shifts the display one character position.

T causes mode to display a test pattern for display alignment. The T parameter cannot be used without the m parameter.

s defines the scroll mode and may be one of the following:
0 Software scroll mode
1 Jump scroll mode
2 Smooth (hardware) scroll mode

The default scroll mode is 0 (software scroll mode)

Press any key to return to the main menu or RETURN to exit:

Durch Eingabe einer **4** können Sie Erklärungen zur Anpassung eines Seriellausganges aufrufen; durch Eingabe einer **5** rufen Sie Erklärungen zur Umleitung eines Parallelausganges auf. Durch Betätigen einer beliebigen Taste kehren Sie zur Übersicht zurück. Auch hier können Sie das Programm durch Eingabe von **RETURN** verlassen.

MODE Version 3.02
Configuring a Serial Device

To Configure a serial device, enter a command line in the form

MODE COMn:baud[,[parity][,[databits][,[stopbits][,P]]]]

where n is 1 or 2 and desognates the specific COM device (Com1: or COM2:) to be configured;
baud is the baud rate (110, 150, 300, 600, 1200, 2400, 4800, or 9600);
parity specifies whether odd (O), even (E), or no (N) parity is to be used;
databits is the world length in bits (7 or 8) exclusive of any parity or stop bits;
stopbits is the number of stop bits (1 or 2) required; and P invokes continuous retry on time-out errors.

Defaults are even parity, 7 data bits, 2 stop bits if the specified baud rate is 110, 1 stop bit if the specified baud rate is not 110, and no continuous retry on time-out errors. There is no default for device nor for the baud rate.

Press any key to return to the main menu or RETURN to exit:

MODE Version 3.02
Remapping Parallel Output

To remap parallel aoutput to a serial port, enter a command line in the form

MODE LPT#:=COMn

where # is 1, 2, or 3 and designates the specific LPT device (LPT1:, LPT2:, or LPT3:) to be remapped
n is 1 or 2 and designates the specific COM device (COM1: or COM2:) to which parallel output is to be mapped.

IMPORTANT: Before you can map parallel output (LPT#:) to a serial port (COMn:), the COM device must be defined with a MODE command line in the form MODE COMn:baud[,[parity][,[databits][,[stopbits][,P]]]]

Anpassung eines Parallelausganges

Möchten Sie mit dem Programm MODE einen Parallelausgang anpassen, geben Sie folgende Befehlszeile ein:

MODE LPT#:[n] [,[m] [,P]

dabei bedeutet:

#	1, 2 oder 3 bestimmt den anzupassenden Ausgang LPT1:, LPT2: oder LPT3:
n	80 oder 132, bestimmt die Zahl der Zeichen pro Zeile
m	6 oder 8, bestimmt die Zahl der Zeichen pro Inch
P	bestimmt, daß der Drucker wiederholt angerufen wird, auch wenn er nicht antwortet (eingeschaltet ist). Diese Funktion kann mit der Tastenfunktion **CTRL-BREAK** beendet werden.

Rufen Sie das Programm MODE zum ersten Mal nach dem Systemstart auf und bestimmen Werte für den Parallelausgang, wird der nicht überschreibbare Teil des Programmes als Erweiterung des Betriebssystemes geladen. Der Systemspeicher vergrößert sich damit um ca. 256 Bytes. Es erscheint die Meldung:

Resident portion of MODE loaded	Nicht überschreibbarer Teil des Programmes MODE geladen

Geben Sie für *n* oder *m* keine Werte ein, werden diese vom Programm eingegeben. Das können entweder die beim Systemstart vorgegebenen Werte sein oder Werte der letzten Eingabe. Beim Systemstart wird der Parallelausgang mit 80 Zeichen pro Zeile und 6 Zeilen pro Inch bestimmt. Antwortet der Drucker nach einer bestimmten Zeitspanne nicht, wird eine Fehlermeldung ausgegeben. Der Vorgang wird dann abgebrochen. Mit Eingabe eines P bestimmen Sie, daß keine Fehlermeldung ausgegeben wird, sondern es wird fortlaufend versucht, Daten zum Drucker auszugeben. Diese unendliche Schleife kann jederzeit durch die Tastenkombination **CTRL-BREAK** beendet werden. Soll die Funktion generell wieder ausgeschaltet werden, müssen Sie den Befehl für MODE erneut ohne den Zusatz P eingeben.

MODE

Sämtliche Eingaben nach dem Befehlswort müssen durch ein Komma getrennt werden. Ein Beispiel:

MODE LPT2:132,8,P

bestimmt für den zweiten Parallelausgang 132 Zeichen pro Druckzeile, 8 Zeilen pro Inch und fortlaufende Versuche, Daten an den Drucker auszugeben. Es werden folgende Meldungen ausgegeben:

LPT2: not redirected	LPT2: nicht umgeleitet
LPT2: set for 132	LPT2: bestimmt mit 132 Zeichen
Printer lines per inch set	Zeilen pro Inch bestimmt
Infinite retry on parallel printer timeout	Der Paralleldrucker wird fortlaufend angerufen

Durch Eingabe eines Kommas (,) bei der Befehlseingabe werden die bereits vorhandenen Werte für einen Ausgang übernommen. Ein Beispiel:

MODE LPT1:,,P

Die Werte für den ersten Parallelausgang werden übernommen, zusätzlich wird die Funktion zur fortlaufenden Datenausgabe aufgerufen für den Fall, daß der Drucker Anfragen der Zentraleinheit nicht beantwortet.

Anpasung des Bildschirmausganges

Das Programm MODE ermöglicht es Ihnen, zwischen einem monochromen und einem graphikfähigen *Bildschirm* umzuschalten. Zusätzlich können für den Graphikschirm verschiedene Betriebsarten ausgewählt werden. Geben Sie den Befehl ein in der Form:

MODE [n] [,[m] [,[t]] [,s]]

dabei bedeutet:

n bestimmt das Darstellungsformat mit folgender Eingabe:

BW40 schaltet die Farbwiedergabe aus, bestimmt 40 Zeichen pro Zeile

BW80 schaltet die Farbwiedergabe aus, bestimmt 80 Zeichen pro Zeile

CO40 schaltet die Farbwiedergabe ein, bestimmt 40 Zeichen pro Zeile

CO80 schaltet die Farbwiedergabe ein, bestimmt 80 Zeichen pro Zeile

GR40 schaltet auf Graphik mit mittlerer Auflösung (320 x 200 Pixel) um, bestimmt 40 Zeichen pro Zeile

GR80 schaltet auf Graphik mit hoher Auflösung (640 x 200 Pixel) um, bestimmt 80 Zeichen pro Zeile

MONO legt die Ausgabe auf den monochromen (einfarbigen) Adapter. Hier werden automatisch 80 Zeichen pro Zeile dargestellt.

m verschiebt die Bildschirmausgabe und ist entweder

R rechts oder

L links

t ruft in der obersten Reihe eine Testzeile auf, die Ihnen hilft, das Bild zu justieren.

s bestimmt den Scrollmodus und hat die Werte

0 Software-Scroll

1 Hardware-Scroll

2 Smooth-Scroll

MODE

Mit der entsprechenden Eingabe für *n* können Sie den Bildschirm auf jeden der sieben möglichen Betriebsarten umschalten. Haben Sie an Ihren Zenith PC über eine Monochromekarte einen IBM-kompatiblen Monochromeschirm angeschlossen, können Sie mit der Eingabe

```
MODE MONO
```

vom Graphikschirm auf diesen Monochromeschirm umschalten. Geben Sie diesen Befehl für einen Graphikschirm ein, erfolgt ebenfalls eine Umschaltung. Sie können dann aber mit Ihrem Gerät nicht mehr arbeiten. Es wird erforderlich sein, das System neu zu starten.

Möchten Sie den Systemstart mit einem angeschlossenen Monochromeschirm vornehmen und nicht mit dem Programm MODE umschalten, muß mit dem Programm SETUP des Monitors ein anderer Wert gespeichert werden. Nähere Informationen hierzu finden Sie in Ihrem Gerätehandbuch. Dies gilt für Geräte der Serie Z-200.

Bei Geräten der Serie Z-100 muß in der Zentraleinheit ein Schalter umgestellt werden. Auch hier finden Sie weitere Einzelheiten in Ihrem Gerätehandbuch.

Geben Sie für *m* entweder ein **R**(echts) oder ein **L**(inks) ein, verschieben Sie das gesamte Bild auf dem Schirm. Zusätzlich sollte die Testzeile aufgerufen werden. Mit ihrer Hilfe kann das Bild zentriert werden. Geben Sie ein

```
MODE ,R, T
```

erscheinen in der obersten Zeile Ziffern und darunter die Frage:

Do you see the rightmost 9?	Sehen Sie die rechte 9?

Ist das Bild richtig zentriert, beantworten Sie die Frage mit einem **Y** für Ja. Geben Sie ein **N**(ein) ein, wird das Bild nach rechts verschoben. Bei einem Schirm mit 40 Zeichen wird das Bild um ein Zeichen verschoben, bei einem Schirm mit 80 Zeichen um zwei Zeichen. Danach erscheint wieder die Testzeile mit der Frage. Verschieben Sie das Bild zu weit nach rechts oder links, arbeitet das System nicht mehr richtig und muß neu gebootet werden.

MODE

Soll nach einer Bildjustierung vom Textmodus in einen Graphikmodus umgeschaltet werden, kann es zu nicht vorhersehbaren Folgen kommen. Sie werden mit folgendem Hinweis darauf aufmerksam gemacht:

WARNING: Changing modes
without shifting the screen
to the right may lock up
your system.
Do you wish to continue? (Y/N)

ACHTUNG: Es besteht die Möglichkeit, daß das System angehalten wird,
wenn Sie den Modus verändern,
ohne das Bild vorher nach rechts
zu verschieben.
Möchten Sie mit dem Programm
fortfahren? (J/N)

Nach der Eingabe von **N**(ein) erfolgt keine Umschaltung, nach Eingabe eines **Y** für Ja wird der eingegebene Befehl ausgeführt. Ist der Bildschirm dann leer und zeigt keinen Text mehr an, muß das System neu gestartet werden.

Der Zusatz T kann nicht ohne einen Wert für *m* eingegeben werden, dagegen kann der Wert für *m* alleine eingegeben werden. Geben Sie mehrere Werte ein, werden diese in der Befehlszeile durch ein Komma getrennt.

Mit dem Wert für *s* bestimmen Sie den SCROLL-Modus. Dieser Modus bestimmt die Art, in der ein *voller Bildschirm* um eine Zeile nach oben geschoben oder gerollt wird. Folgende Modi sind verfügbar:

0 Software-Scroll. Hierbei wird der Schirmtext tatsächlich auf dem Bildschirm bewegt. Ohne weitere Eingabe ist dieser Modus vorgegeben.

1 Hardware-Scroll. Dieser Modus ist schneller als der Software-Scroll. Allerdings ist er im Gegensatz zu diesem nicht zu allen Anwendungspaketen kompatibel. Hardware-Scrolling ist nur bei 80 x 25 Zeichen und im Grafikmodus anwendbar.

2 Smooth-Scroll (weiches Rollen). Dieser nur bei hochauflösender Grafik verfügbare Modus bewirkt ein besonders gut lesbares Rollen. Erreicht wird dies dadurch, daß die Zeilen in kleinen Schritten und nicht zeilenweise bewegt werden.

Der SCROLL-Modus 0 ist zu allen Bildschirmmodi und Anwenderprogrammen kompatibel. Die Modi 1 und 2 können nur mit Einschränkungen benutzt werden. Die Tabelle 11.5 gibt eine Übersicht.

Tabelle 11.5 Gültiger SCROLL-Modus

BILD-MODUS	SOFTWARE	HARDWARE	SMOOTH
BW40/CO40	gültig	ungültig	ungültig
BW80/CO80	gültig	gültig	ungültig
GR40	gültig	gültig	gültig
GR80	gültig	gültig	gültig
MONO	gültig	ungültig	ungültig

Anpassung eines Seriellausganges

Möchten Sie mit dem Programm MODE einen *seriellen Ausgang* für den Anschluß eines Peripheriegerätes (z. B. Drucker oder Modem) anpassen, geben Sie folgende Befehlszeile ein:

MODE COMn:Baud-Rate [Parity] [,[Wort-Länge] [,Stop-Bit] [,P]]]

dabei bedeutet:

n	1 oder 2, bestimmt den anzupassenden Seriellausgang
Baud-Rate	110, 150, 300, 600, 1200, 2400, 4800 oder 9600, bestimmt die Übertragungsgeschwindigkeit (Baud-Rate)
Parity	O (Odd), E (Even) oder N (No). Bestimmt die Art der Parity
Wort-Länge	7 oder 8, bestimmt die Wortlänge (Data-Bits, errechnet ohne Stop- oder Prüf-Bit)
Stop-Bit	1 oder 2, bestimmt die Anzahl der erforderlichen Stop-Bits
P	bestimmt, daß der Drucker wiederholt angerufen wird, auch wenn er nicht antwortet (eingeschaltet ist). Diese Funktion kann mit der Tastaturkombination **CTRL-BREAK** beendet werden.

Rufen Sie das Programm MODE zu ersten Mal nach dem Systemstart auf und bestimmen Werte für den *Seriellausgang*, wird der nicht überschreibbare Teil des Programmes als Erweiterung des Betriebssystemes geladen. Der Systemspeicher vergrößert sich damit um ca. 256 Bytes. Es erscheint die Meldung:

Resident portion of MODE loaded	Nicht überschreibbarer Teil des Programmes MODE geladen

Bei der Befehlseingabe muß immer der Ausgang (COM1: oder COM2:) bestimmt werden, ebenso der Wert für die Übertragungsgeschwindigkeit. Wird ein serieller Drucker an den zu definierenden Ausgang angeschlossen, ist zusätzlich der Wert für P einzugeben. Alle anderen Werte werden bei Nichteingabe vom Programm vorgegeben.

Dies sind entweder die Werte des letzten Befehls oder die im Programm MODE gespeicherten Werte. Im Programm sind folgende Werte gespeichert:

- even Parity
- 7 Data-Bits
- 2 Stop-Bits für eine Baud-Rate von 110, 1 Stop-Bit für alle anderen Werte
- keine wiederholten Anrufe des Gerätes bei Zeitüberschreitungsfehlern

Sämtliche Eingaben müssen in der Befehlszeile durch ein Komma getrennt werden. Ein Beispiel:

```
MODE COM1:1200,E,7,2
```

Für den ersten seriellen Ausgang wird die Übertragungsgeschwindigkeit (Baud-Rate) mit 1200 bestimmt. Es werden 1200 Binärzeichen pro Sekunde zwischen der Zentraleinheit und dem angeschlossenen Peripheriegerät übertragen. Die Parität ist gerade (Even). Die übertragenen Zeichen bestehen aus 7 Daten-Bits und werden mit 2 Stop-Bits abgeschlossen.

Sollen bereits gespeicherte Werte übernommen werden, geben Sie für diese Werte nur das Komma ein. Ein Beispiel:

```
MODE COM1:2400,,,P
```

Für den ersten seriellen Ausgang wird jetzt eine Baud-Rate von 2400 Zeichen pro Sekunde bestimmt und fortlaufende Versuche, Daten an das Gerät auszusenden. Die Werte für Parity, Daten-Bits und Stop-Bits werden unverändert übernommen.

Der Wert für Baud-Rate kann auch in einer Kurzform eingegeben werden. Dafür sind mindestens zwei Ziffern erforderlich. Ein Beispiel:

```
MODE COM1:24
```

Mit diesem Befehl werden alle vorgegebenen Werte für den Ausgang COM1: übernommen. Geändert hat sich nur der Wert P. Die Funktion zum wiederholten Anruf des Gerätes ist jetzt ausgeschaltet.

Umleitung eines Parallelausganges

Soll ein serieller Drucker an Ihren Zenith PC angeschlossen werden, ist es unbedingt erforderlich, den *Parallelausgang* zum seriellen Ausgang umzuleiten. Dies geschieht mit dem Befehl

MODE LPT#:=COMn

dabei bedeutet:

#	1, 2 oder 3, bestimmt den umzuleitenden Parallelausgang (LPT1:, LPT2: oder LPT3:)
n	1 oder 2, bestimmt den Seriellausgang, zu dem die Daten gesendet werden (COM1: oder COM2:)

Vorher muß das Protokoll des seriellen Ausganges bestimmt werden. Dies geschieht mit dem Befehl MODE in der Form:

MODE COMn:Baud-Rate [,[Parity] [,[Daten-Bits] [,[Stop-Bits] [,P]]]]

Wird ein Drucker angeschlossen, muß bei der Bestimmung des Protokolls für den seriellen Ausgang unbedingt der Wert P mit eingegeben werden. Soll z. B. an den Ausgang COM2 ein Drucker angeschlossen werden, muß erst das Übertragungsprotokoll für den Ausgang bestimmt werden. Dann muß der parallele Ausgang zu diesem Ausgang umgeleitet werden. Dies geschieht mit folgendem Befehl:

MODE LPT1:=COM2

Eine *Umleitung* zu einem seriellen Ausgang wird gelöscht, sobald die Werte für diesen Parallelausgang neu bestimmt werden. Dies ist möglich mit einem Befehl in der Form:

MODE LPT#: [n] [,m] [,P]

Nähere Erläuterungen zu dieser Befehlszeile finden Sie im Abschnitt: Anpassung eines Parallelausganges.

Fehlermeldungen

Illegal device name — Falscher Name für das Peripheriegerät

ERKLÄRUNG:
Sie haben für den Parallelausgang (LPT) oder für den Seriellausgang (COM) einen falschen Namen eingegeben. Wiederholen Sie die Eingabe mit einem gültigen Namen.

Invalid baud rate specified — Falsche Baud-Rate eingegeben

ERKLÄRUNG:
Sie haben einen falschen Wert für die Übertragungsgeschwindigkeit eingegeben. Wiederholen Sie die Eingabe mit einem gültigen Wert.

Invalid parameters — Falsche Zusatzeingabe

ERKLÄRUNG:
Sie haben in der Befehlszeile falsche Werte eingegeben. Diese Fehlermeldung erscheint auch, wenn Sie in der Befehlszeile nicht alle erforderlichen Werte eingeben.

MORE   Extern

Zweck

gibt Daten seitenweise auf dem Bildschirm aus

Syntax

[d:] [Weg] MORE
Befehl ¦ MORE

dabei bedeutet:

d:	vor dem Programmnamen bestimmt das aufzurufende Laufwerk
Weg	vor dem Programmnamen bestimmt das aufzurufende Verzeichnis

Bemerkungen

Das Programm MORE kann im Netzwerk ohne Einschränkungen benutzt werden.

Mit diesem Filter können Sie größere Datenmengen seitenweise auf den Bildschirm ausgeben. Am Ende einer jeden Seite erscheint die Zeile

-- MORE --   -- Weiter --

und die Ausgabe wird unterbrochen. Erst nach Eingabe von **RETURN** wird die Ausgabe fortgesetzt. Wie jeder andere Filter auch liest MORE erst Daten ein, bevor es die Daten anzeigt. Die Eingabe erfolgt über das Standardeingabegerät.

Ein Beispiel:

TYPE BRIEF.TXT ¦ MORE

Der Befehl TYPE zeigt den Inhalt der Datei BRIEF.TXT auf dem Bildschirm an. Ist es eine größere Datei, wird die Ausgabe nach jeder vollen Seite angehalten und erst durch Betätigen der Taste **RETURN** fortgesetzt.

Da von MORE die Daten über das Standardeingabegerät gelesen werden, ist auch folgende Kombination möglich:

MORE < BRIEF.TXT

Sie können jetzt, falls gewünscht, einzelne Seiten der Datei mit der Tastenkombination **SHIFT-PRTSC** auf den Drucker ausgeben. Nähere Einzelheiten dazu im Kapitel 8.

Fehlermeldung

Invalid DOS version — Falsche DOS-Version

ERKLÄRUNG:
Sie benutzen den Filter MORE unter einer DOS-Version, die nicht 2.0 oder höher ist.

NODEBUG Extern

Zweck

schaltet das interne Monitorprogramm aus

Syntax

[d:] [Weg] NODEBUG

dabei bedeutet:

d: vor dem Programmnamen bestimmt das aufzurufende Laufwerk

Weg vor dem Programmnamen bestimmt das aufzurufende Verzeichnis

Bemerkungen

Das Programm NODEBUG kann im *Netzwerk* **ohne** Einschränkungen benutzt werden.

Fast alle für den IBM-PC und andere kompatible Rechner geschriebenen Programme arbeiten auch auf Ihrem Zenith PC einwandfrei. Trotzdem kann es vereinzelt vorkommen, daß ein Programm nicht sofort *fehlerfrei* arbeitet. Dies liegt an einer Besonderheit Ihres Gerätes. Im ROM ist ein Monitorprogramm gespeichert, daß auch verschiedene Funktionen zur Fehlersuche beinhaltet. Dieses Monitorprogramm kann jederzeit von Ihnen durch eine Tastenkombination aufgerufen werden (s. a. Kapitel 16). Wird diese Kombination von einem Anwenderprogramm aufgerufen, hat auch dies zur Folge, daß Sie das normale Programm verlassen und sich im Monitor befinden. Mit dem Programm NODEBUG, das auf der zweiten Systemdiskette enthalten ist, können Sie das Monitorprogramm ausschalten. Rufen Sie das Programm auf durch Eingabe von:

NODEBUG

NODEBUG

Ruft ein Anwenderprogramm den Monitor auf, werden sämtliche Registerwerte des Prozessors in der folgenden Form angezeigt:

```
AX=0000 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=351D ES =351D SS=351D CS=351D IP=0100  NV UP DI PL NZ NA PO NC
351D:0100 FE0F         DEC         BYTE PTR [BX]
->
```

Um den Monitor jetzt auszuschalten, starten (booten) Sie das Betriebssystem mit der Tastenkombination **CTRL-ALT-DEL** neu. Nach den Anfragen für Datum und Uhrzeit schalten Sie dann mit Eingabe von

NODEBUG

den Monitor aus. Haben Sie festgestellt, daß ein Programm laufend den Monitor aufruft, sollten Sie dieses Programm in Zukunft über eine Batchdatei aufrufen. Die Batchdatei ruft zuerst das Programm NODEBUG auf und dann Ihr Anwendungsprogramm. Weitere Erklärungen zum Erstellen einer solchen Datei finden Sie im Kapitel 5.

PATH Intern

Zweck

benennt Wege, die zu einem Verzeichnis führen, in dem ein externer Befehl gesucht werden soll.

Syntax

PATH [D:] [Weg [;[D:] Weg] ...]

dabei bedeutet:

D:	Name des Laufwerkes, auf dem der Weg existiert
Weg	mehrere Verzeichnisse in Folge in der Form: [\] [Verzeichnis [\Verzeichnis ...]]

Bemerkungen

Rufen Sie einen externen Befehl auf, sucht das Betriebssystem dieses Programm (Befehl) immer im momentanen Inhaltsverzeichnis. Wird das Programm nicht gefunden, erscheint die Fehlermeldung:

Bad command or file name	Befehl oder Datei falsch

Arbeiten Sie z. B. auf einem Winchesterlaufwerk, ist es vorteilhaft, die Dateien in verschiedenen Verzeichnissen zu speichern. Dadurch wird das einzelne Verzeichnis kleiner und kann schneller abgesucht werden. Damit trotzdem noch auf Programme oder Befehle zugegriffen werden kann, die in einem separaten Verzeichnis gespeichert sind, muß MS-DOS wissen, in welchem Verzeichnis diese Programme zu suchen sind. Dies wird mit dem Befehl PATH bestimmt.

Geben Sie den Befehl ohne Zusätze ein:

PATH

erscheint nach dem Systemstart folgende Meldung:

No path

PATH

Es kann von MS-DOS erst dann auf einen *Weg* zurückgegriffen werden, nachdem Sie entsprechende Verzeichnisse mit dem Befehl PATH eingegeben haben.

Geben Sie den Weg ein, indem Sie nach dem Befehlsnamen ein oder mehrere Verzeichnisse benennen. Stellen Sie dem ersten Verzeichnis einen Schrägstrich (\) voran, geben Sie einen absoluten Weg ein. Dieser Weg beginnt dann immer im Rootdir. Ohne den Schrägstrich beginnt der Weg im momentanen Verzeichnis. Dies ist dann ein relativer Weg. Nähere Informationen dazu im Kapitel 7.

Haben Sie z. B. alle Systemdateien im Verzeichnis SYSTEM gespeichert und dieses Verzeichnis vom Rootdir aus angelegt, lautet der absolute Weg zu diesem Verzeichnis:

\SYSTEM

Geben Sie diesen Weg in der folgenden Form ein:

```
PATH C:\SYSTEM
```

Rufen Sie jetzt den Befehl ohne Zusatz auf, wird angezeigt:

```
PATH=C:\SYSTEM
```

Sie haben auch die Möglichkeit, mehrere Wege zugleich einzugeben. Dann werden die einzelnen Wege durch ein Semikolon (;) getrennt. Diese Eingabe wird nur durch den Tastaturspeicher auf 128 Zeichen begrenzt. Ein Beispiel:

```
PATH C:\SYSTEM;D:\PROGRAMM\TEXT\DEV
```

Gespeicherte Wege werden bei jeder Neueingabe überschrieben, so daß immer die letzte Eingabe gespeichert ist. Sollen alle Wege gelöscht werden, geben Sie ein:

```
PATH;
```

Rufen Sie jetzt den Befehl ohne weitere Eingaben auf, erscheint:

```
No path
```

PATH

Fehlermeldung

Out of environment space — Kapazität des Speichers erschöpft

ERKLÄRUNG:
Vom System wird ein Zwischenspeicher von 200 Bytes benutzt, um diese Werte zu speichern. Der Speicher ist voll. Löschen Sie nicht benötigte Werte mit dem Befehl SET.

PAUSE — Intern

Zweck

unterbricht die Abarbeitung einer Batchdatei und gibt einen Text auf dem Bildschirm aus

Syntax

PAUSE [Text]

Bemerkungen

Mit diesem Befehl können Sie jederzeit die Abarbeitung einer Batchdatei vorübergehend *anhalten.* Sie haben zusätzlich die Möglichkeit, einen eigenen Text auszugeben. Dies ist z. B. wünschenswert, wenn der Benutzer eine Diskette wechseln soll. Der Befehl Pause gibt einmal Ihren Text aus und schreibt dann in die nächste Zeile die Worte:

Strike a key when ready ...	Betätigen Sie eine Taste, wenn Sie fertig sind ...

Jetzt wartet das Programm auf Ihre Eingabe. Sie können irgendeine beliebige alphanumerische Taste betätigen, damit das Programm fortfährt. Betätigen Sie die Tastenkombination **CTRL-C,** erscheint die Frage:

Terminate batch job (Y/N)?	Wollen Sie die Batchdatei verlassen (J/N)?

Geben Sie jetzt ein **Y** für Ja ein, verlassen Sie die Batchdatei und sind wieder im Betriebssystem. Geben Sie ein **N**(ein) ein, wird die Datei normal weiter abgearbeitet.

Haben Sie eine Batchdatei geschrieben, in der z. B. die Diskette gewechselt werden muß, können Sie den Befehl PAUSE folgendermaßen eingeben:

...

...

PAUSE Legen Sie in das Laufwerk B eine neue Diskette.

...

...

PAUSE

Die Batchdatei wird bis zum Befehl PAUSE abgearbeitet, dann erscheint auf dem Bildschirm folgende Meldung:

```
Legen Sie in das Laufwerk B eine neue Diskette.
Strike a key when ready ...
```

Ein weiteres Beispiel:

In der Batchdatei soll an einer bestimmten Stelle der Bediener eventuell die Datei verlassen können. Auch hier bietet sich der Befehl PAUSE an:

```
Durch Eingabe der Tastenkombination <CTRL-C> verlassen Sie die Datei.
Strike a key when ready ...
```

Betätigen Sie jetzt die Tasten **CTRL-C**, erscheint die Frage:

```
Terminate batch job (Y/N)?
```

Durch Eingabe von **Y** sind Sie wieder im Betriebssystem.

PRINT **Extern**

Zweck

gibt eine ASCII-Datei auf den Drucker aus, während ein anderes Programm auf dem Bildschirm bearbeitet wird

Syntax

[d:] [Weg] PRINT
[d:] [Weg] PRINT Dateibez [/X] [Dateibez [/X] ...]

dabei bedeutet:

d: vor dem Programmnamen bestimmt das aufzurufende Laufwerk

Weg vor dem Programmnamen bestimmt das aufzurufende Verzeichnis

Dateibez Name der zu druckenden Datei

/X Kombination aus folgenden Zusatzeingaben:

- **/A** Abbruch, löscht die bezeichnete Datei der Liste und alle folgenden Dateien der Befehlszeile bis zum Zusatz /S.
- **/B:*n*** Speichergröße, bestimmt die Größe des internen Zwischenspeichers für die zu druckende Datei. n ist ein Wert zwischen 512 und 16 384.
- **/C:*n*** Kopie, druckt die Datei n-mal. n ist eine Zahl von 1 bis 255.
- **/D:*dev*** Ausgabegerät, bestimmt das Peripheriegerät, zu dem die Daten gesendet werden sollen. Ohne weitere Eingabe ist dies PRN (Systemdrucker).
- **/F** Seitenvorschub, fügt ein leeres Blatt zwischen die zu druckenden Dateien ein.

/L:*n* linker Rand, setzt den linken Rand an die n-te Spalte.

/M:*n* maximale Druckdauer, bestimmt die Anzahl der Prozessortakte, die ohne Unterbrechung gedruckt wird. n ist eine Zahl von 1 bis 255

/N:*n* maximale Druckpause, bestimmt die Anzahl der Prozessortakte, die bis zum nächsten Druck mindestens verstreichen muß. n ist eine Zahl von 1 bis 255.

/P:*n* Seitenlänge, bestimmt die Zeilenzahl der Seite mit n-Zeilen für diese Datei.

/Q:*n* Größe der Liste, bestimmt die maximale Zahl der Dateien in der Druckliste. n ist eine Zahl von 1 bis 32.

/R:*n* rechter Rand, setzt den rechten Rand an die n-te Spalte.

/S weiter, gibt die bezeichnete Datei und alle folgenden der Befehlszeile in die Warteschlange ein bis zum Zusatz /A.

/T Ende, löscht alle Dateien der Warteschlange und beendet das Druckprogramm.

/U:*n* Zeitfehler, bestimmt die maximale Wartezeit des Programmes PRINT auf eine Antwort des Druckers. n ist eine Zahl von 1 bis 255.

Bemerkungen

Das Dienstprogramm PRINT *druckt* die Dateien zur gleichen Zeit, während Sie andere Programme bearbeiten. Sie arbeiten im Vordergrund (Bildschirm), das Programm arbeitet im *Hintergrund* (Drucker). Dies ist möglich, da Sie den Prozessor nur jeweils für sehr kurze Zeit benötigen. D. h., der Prozessor verbringt viel Zeit mit Warten, weil z. B. Text auf dem Bildschirm ausgegeben wird. In dieser Zeit kann der Prozessor andere Aufgaben wahrnehmen. In diesem Fall kann er Daten zum Drucker senden. Diese Arbeitsweise wird Hintergrundverarbeitung genannt.

Sie geben die zu druckenden Dateien in eine Warteliste ein. Diese kann maximal 32 Dateinamen speichern. Die Dateien werden dann nacheinander aus der Liste zum Drucken abgerufen. Sie haben jederzeit die Möglichkeit, Dateien der Schlange zu löschen oder neue hinzuzufügen.

Dateiaufruf

Geben Sie den Programmnamen ohne weitere Zusätze ein, werden die Dateien der Warteliste angezeigt. Ist die Liste leer, erscheint die Meldung:

PRINT queue is empty	PRINT-Warteliste ist leer

Die zu druckenden Dateien müssen hinter dem Befehlsnamen, getrennt durch ein Leerfeld, eingegeben werden. Dabei kann für jeden Dateinamen eine Laufwerkbezeichnung und/oder ein Verzeichnisname vorangestellt werden. Die Zusatzeingaben müssen direkt nach der Datei erfolgen, auf die sie sich beziehen. Geben Sie Zusatzeingaben nach der letzten Datei der Zeile ein, beziehen sich diese Eingaben auch nur auf die letzte Datei. Sie gelten nicht für vorherige Dateinamen. Geben Sie mehr Dateinamen ein, als die Warteliste speichern kann, erscheint die Meldung:

PRINT queue is full	PRINT-Schlange ist voll

Sie haben die Möglichkeit, mehrere Dateinamen mit dem Stern oder dem Fragezeichen zusammenzufassen. Geben Sie nach diesen Dateinamen Zusätze ein, gelten diese Zusätze für alle zusammengefaßten Dateinamen.

Zusatzeingaben

Mit 14 Zusatzeingaben beeinflussen Sie die Arbeitsweise des Programmes PRINT. Bis auf die Zusätze /A und /S gelten sie nur jeweils für die vorangehende Datei. Die Zusätze /A und /S gelten für alle Dateien der Eingabezeile solange, bis jeweils der andere Zusatz eingegeben wird.

Sechs Zusatzeingaben können nur beim ersten Aufruf des Programmes PRINT benutzt werden. Es sind dies die Zusätze: /B, /D, /M, /N, /Q und /U. Hiermit beeinflussen Sie die generelle Arbeitsweise des Programmes. Sie müssen diese Zusätze direkt nach dem Programmnamen in der Befehlszeile eingeben. Danach können dann andere Eingaben erfolgen.

/A -- Abbruch

Die bezeichnete Datei wird aus der Warteschlange *gelöscht*. Werden nach dem Zusatz weitere Dateinamen eingegeben, werden auch diese gelöscht. Der Zusatz gilt solange für Dateinamen der Befehlszeile, bis Sie den Zusatz /S eingeben.

/B:n -- Speichergröße

Mit diesem Zusatz bestimmen Sie die Größe des internen Zwischenspeichers für die zu druckenden Dateien. Ohne Eingabe kann dieser Speicher 512 Zeichen aufnehmen. Mit dem Wert für n können Sie eine maximale Größe von 16 384 Bytes bestimmen. Damit kann die Arbeitsweise des Programmes beschleunigt werden, da weniger Plattenzugriffe erforderlich werden.

Dieser Zusatz darf nur beim ersten Aufruf des Programmes PRINT eingegeben werden. Später wird er nicht mehr beachtet. Soll der Wert geändert werden, müssen Sie das System neu starten.

/C:n -- Kopien

Das Programm PRINT druckt jede Datei einmal. Der Zusatz bestimmt die Anzahl der *Kopien* dieser Datei. n ist eine Zahl zwischen 1 und 255. Soll z. B. die Datei dreimal gedruckt werden, geben Sie nach dem Dateinamen ein:

```
C3
```

/D:dev -- Ausgabegerät

Bestimmen Sie mit diesem Zusatz das Peripheriegerät, zu dem die Daten gesendet werden sollen. Geben Sie nach dem Doppelpunkt einen gültigen Namen für ein vorhandenes Gerät an. Eine Übersicht finden Sie im Kapitel 1. Ein Beispiel:

```
PRINT /D:COM1
```

Das Ausgabegerät ist ein serieller Drucker, der am Ausgang COM1: angeschlossen ist. Benutzen Sie diesen Zusatz nicht, werden Sie vom Programm gefragt:

```
Name of list device [PRN:]
```

Auch jetzt können Sie ein anderes Gerät bestimmen als den Systemdrucker (PRN). Mit der Eingabe von RETURN übernehmen Sie den vorgeschlagenen Namen.

Dieser Zusatz darf nur beim ersten Aufruf des Programmes PRINT eingegeben werden. Später wird er nicht mehr beachtet. Soll der Wert geändert werden, müssen Sie das System neu starten.

/F -- Blattvorschub

Dieser Zusatz veranlaßt den Drucker, nach der bezeichneten Datei einen Blattvorschub auszuführen. Die folgende Datei wird dann zu Beginn einer neuen Seite gedruckt. Dies gilt auch, wenn von einer Datei mehrere Kopien gedruckt werden sollen.

/L:n -- linker Rand

Normalerweise beginnt das Programm in der ersten Spalte mit dem Druck der Datei. Durch Eingabe dieses Zusatzes kann mit *n* die Anfangsspalte bestimmt werden. Mit n kann jede Spalte Ihres Druk- kers bestimmt werden. Die gesamte Datei wird dann beginnend ab dieser Spalte gedruckt. Der Zusatz gilt nur für die voranstehende Datei.

/M:n -- maximale Druckdauer

Mit diesem Zusatz bestimmen Sie die maximale Druckzeit des Programmes PRINT, während das Vordergrundprogramm unter- brochen ist. Die Dauer wird in Prozessortakten gemessen. Der Prozessor Ihres Zenith PC wird mit 4 700 000 oder 8 000 000 Tak- ten pro Sekunde betrieben. Ohne Eingabe von /M ist ein Wert von 2 Takten vorgegeben. Vergrößern Sie diesen Wert, wird zwar der Druck schneller ausgeführt, aber gleichzeitig arbeitet das Vorder- grundprogramm langsamer. Sie können einen Wert zwischen 1 und 255 bestimmen.

Dieser Zusatz darf nur beim ersten Aufruf des Programmes PRINT eingegeben werden. Später wird er nicht mehr beachtet. Soll der Wert geändert werden, müssen Sie das System neu starten.

/N:n -- maximale Druckpause

Mit diesem Zusatz bestimmen Sie die maximale Druckpause des Programmes PRINT, während der das Vordergrundprogramm nicht unterbrochen wird. Die Dauer wird in Prozessortakten gemessen. Der Prozessor Ihres Zenith PC wird mit 4 700 000 oder 8 000 000 Takten pro Sekunde betrieben. Ohne Eingabe von /N ist ein Wert von 8 Takten vorgegeben. Vergrößern Sie diesen Wert, wird zwar das Vordergrundprogramm schneller ausgeführt, aber gleichzeitig dauert der Druck länger. Sie können einen Wert zwischen 1 und 255 bestimmen. Dieser Zusatz darf nur beim ersten Aufruf des Programmes PRINT eingegeben werden. Später wird er nicht mehr beachtet. Soll der Wert geändert werden, müssen Sie das System neu starten.

Das Verhältnis der beiden Zusätze /M und /N bestimmt den Prozentsatz der CPU-Zeit, die das Programm PRINT erhält. Dies läßt sich mit folgender Formel errechnen:

PRINT-ZEIT = max. Druckdauer / (1 + max. Druckpause) * 100

Die vorgegebenen Werte (2 für maximale Dauer und 8 für maximale Pause) teilen dem Programm PRINT ca. 25% der CPU-Zeit zu.

Dieser Zusatz darf nur beim ersten Aufruf des Programmes PRINT eingegeben werden. Später wird er nicht mehr beachtet. Soll der Wert geändert werden, müssen Sie das System neu starten.

/P:n -- Seitenlänge

PRINT druckt die Dateien zeilenweise aus, ohne den Text zu formatieren. Mit dem Zusatz /P können Sie die *Zeilenzahl* einer Seite bestimmen. n legt die Anzahl der Zeilen einer Seite fest. Nach der letzten Zeile werden entsprechend Leerzeilen eingefügt, damit die nächste Zeile auf einer neuen Seite beginnt. Enthält der Text selber einen Befehl für einen Seitenumbruch, wird dieser Befehl durch den Zusatz nicht unterdrückt.

/Q:n -- Größe der Warteliste

Mit diesem Zusatz können Sie die Größe der Warteliste bestimmen. Geben Sie den Zusatz nicht ein, speichert das Programm PRINT maximal 10 Dateien. Mit dem Wert für n bestimmen Sie eine Größe zwischen 1 und 32.

Dieser Zusatz darf nur beim ersten Aufruf des Programmes PRINT eingegeben werden. Später wird er nicht mehr beachtet. Soll der Wert geändert werden, müssen Sie das System neu starten.

/R:n -- rechter Rand

Normalerweise druckt das Programm die Zeilen bis zum Zeichen für RETURN in einer Zeile. Durch Eingabe des Zusatzes /R kann die äußerste *rechte Spalte* bestimmt werden. Dies kann maximal die Spalte 255 sein. Geben Sie hier den entsprechenden Wert für Ihren Drucker ein. Das Programm fügt dann ein RETURN ein und druckt den weiteren Text in der nächsten Zeile. Im Text vorhandene RETURNs werden immer zuerst ausgeführt.

/S -- weitere Druckdateien

Mit diesem Zusatz geben Sie weitere Dateien in die Warteliste ein. Sämtliche Dateien der Befehlszeile werden in die Liste gegeben. Diesen Zusatz brauchen Sie nicht beim erstmaligen Aufruf des Programmes mit eingeben. Er wird erst benötigt, wenn schon Dateien in der Warteliste vorhanden sind. Die Dateien werden immer in der Reihenfolge ihrer Eingabe ausgedruckt.

/T -- Ende

Durch diesen Zusatz werden alle Dateien der Warteliste gelöscht und das Programm beendet. Es erscheint die Meldung:

PRINT queue is empty	PRINT-Schlange ist leer

Das Programm kann danach jederzeit wieder durch Eingabe von Dateien in die Warteliste neu gestartet werden. Die Ausgabe erfolgt dann auf das schon vorher bezeichnete Peripheriegerät. Soll dieses geändert werden, müssen Sie das System neu starten.

/U:n -- Zeitfehler

Mit diesem Zusatz können Sie die Wartezeit für PRINT bestimmen, in der das Programm nach einem Druckeranruf auf Antwort wartet. Wird dieser Wert überschritten, gibt PRINT die Kontrolle sofort wieder an das Betriebssystem zurück, und das laufende Programm kann weiterarbeiten. Sie benötigen diesen Zusatz nur für langsame Drucker. Ohne Eingabe von /U ist ein Wert von 1 vorgegeben. Vergrößern Sie diesen Wert, wird zwar der Druck schneller ausgeführt, aber gleichzeitig arbeitet das Vordergrundprogramm langsamer. Sie können einen Wert zwischen 1 und 255 bestimmen.

Dieser Zusatz darf nur beim ersten Aufruf des Programmes PRINT eingegeben werden. Später wird er nicht mehr beachtet. Soll der Wert geändert werden, müssen Sie das System neu starten.

Beispiele

Für das Beispiel sollen folgende Textdateien gedruckt werden:

LETTER.CTR
MYFILE
WOM.BAT
REPORT.DOC

Von der Datei LETTER.CTR sollen zwei Kopien erstellt werden. Nach jeder Datei soll auf einer neuen Seite begonnen werden. Sämtliche Dateien sind im Laufwerk B: gespeichert. Rufen Sie das Programm mit folgender Eingabe auf:

```
PRINT B:LETTER.CTR /C2 /F B:MYFILE /F B:WOM.BAT /F B:REPORT.DOC /F
```

Ist dies der erste Aufruf von PRINT, erscheint folgende Meldung:

Name of list device PRN:	Name des Ausgabegerätes PRN:

PRINT

Geben Sie hier den Namen des Gerätes ein, auf den die Dateien gedruckt werden sollen. Der vorgegebene Wert ist PRN. Diesen übernehmen Sie durch Betätigen der Taste **RETURN.** (Im Kapitel 8 finden Sie eine Liste der möglichen Ausgabegeräte.) Es erscheint jetzt die Meldung

Resident part of PRINT installed	Nicht überschreibbarer Teil des Programmes PRINT geladen

sowie eine Liste der zu druckenden Dateien:

```
B:LETTER .CTR is currently being printed
B:MYFILE .    is in queue
B:WOM    .BAT is in queue
B:REPORT .DOC is in queue
```

Die erste Datei wird bereits gedruckt. Alle weiteren Dateien sind zum Druck bestimmt. Ist der Druck der ersten Datei beendet, wird sie aus der Liste gelöscht und alle folgenden Dateien rücken nach. Geben Sie während des Druckvorganges den Namen des Programmes ohne Zusätze ein, werden die restlichen Dateien der Liste angezeigt.

Durch die Zusätze /A und /S können Sie jetzt die Dateien der Liste löschen oder neue hinzufügen. Es sollen z. B. zwei weitere Dateien hinzugefügt werden, geben Sie ein:

```
PRINT A:FORGOT1.DOC /S/F B:FORGOT2.DOC /F
```

Die beiden Dateien werden in die Warteliste gegeben, und der neue Inhalt wird angezeigt:

```
B:MYFILE  .    is currently being printed
B:WOM     .BAT is in queue
B:REPORT  .DOC is in queue
B:FORGOT1.DOC  is in queue
B:FORGOT2.DOC  is in queue
```

Ebenso können Sie auch die Datei WOM.BAT aus der Liste löschen. Geben Sie ein:

```
PRINT B:WOM.BAT /A
```

Es erscheint jetzt die Meldung:

WOM.BAT Cancelled by operateur	WOM.BAT durch den Bediener gelöscht

Löschen Sie eine Datei, die gerade gedruckt wird, wird der Druck sofort abgebrochen und diese Meldung ausgegeben.

Mit dem Zusatz /T löschen Sie alle noch vorhandenen Dateien der Liste. Gleichzeitig können Sie aber auch neue Dateien zum Druck bestimmen. Der Befehl

```
PRINT /T *.ASM
```

löscht alle Dateien der Liste. Anschließend werden die Dateien des momentanen Laufwerks mit dem Namenszusatz .ASM neu in die Liste eingegeben. Es werden maximal 10 Dateien berücksichtigt.

Fehlermeldungen

Invalid Parameter	Falsche Zusatzeingabe

ERKLÄRUNG:
Sie haben bei einem erneuten Aufruf des Programmes einen Zusatz eingegeben, der nur während des ersten Aufrufes bearbeitet werden kann. Dies gilt für die Zusätze //, /D, /M, /N, /Q und /U. Überprüfen Sie Ihre Eingabe.

PRINT

All files cancelled by operator	Alle Dateien vom Bediener gelöscht

ERKLÄRUNG:
Diese Meldung erscheint, wenn Sie während des Programmes den Zusatz /T eingeben.

D:... Cancelled by operator	D: Datei vom Bediener gelöscht

ERKLÄRUNG:
Diese Meldung erscheint, wenn Sie eine einzelne Datei mit dem Zusatz /A löschen.

Drive not ready	Laufwerk nicht fertig

ERKLÄRUNG:
Diese Meldung erscheint, wenn das Programm versucht, auf die Platte zuzugreifen. Dieser Vorgang wird solange wiederholt, bis ein weiterer Fehler auftaucht. Danach erscheint die Meldung:

PRINT ... Cancelled	PRINT Dateiname gelöscht

PRINT ... File not found	PRINT Datei nicht gefunden

ERKLÄRUNG:
Die bezeichnete Datei ist nicht vorhanden. Überprüfen Sie Ihre Eingabe und geben Sie den Namen neu ein.

Invalid value for /X ignored	Falsche Zusatzeingabe /X nicht beachtet

ERKLÄRUNG:
Sie haben einen falschen Zusatz eingegeben. Dieser wird bei der Abarbeitung des Programmes nicht beachtet. Das Programm wird korrekt weiter abgearbeitet.

List output not assigned to a device	Eingegebener Name bezeichnet kein Ausgabegerät

ERKLÄRUNG:
Sie haben auf die Frage

Name of list device PRN:	Name des Ausgabegerätes PRN:

einen falschen Namen eingegeben. Die Tabelle 8.1 enthält eine Übersicht der gültigen Namen.

Name of list device PRN:	Name des Ausgabegerätes PRN:

ERKLÄRUNG:
Diese Meldung erscheint, wenn Sie das Programm PRINT das erste Mal aufrufen. Mit der Antwort benennen Sie das Periepheriegerät, zu dem die Daten gesendet werden sollen. Betätigen Sie nur die Taste **RETURN,** übernehmen Sie den vorgegebenen Wert (PRN). Die Tabelle 8.1 enthält eine Übersicht der gültigen Namen. Haben Sie einen gültigen Namen eingegeben, erscheint die Meldung:

Resident part of PRINT installed	Nicht überschreibbarer Teil des Programmes PRINT geladen

No files match ...	Keine Datei entspricht ...

ERKLÄRUNG:
Sie haben eine weitere Datei zum Druck bestimmt. Das Programm kann diese Datei nicht finden. Überprüfen Sie die Eingabe.

Löschen Sie jedoch eine zum Druck bestimmte Datei und diese Datei ist nicht vorhanden, erfolgt keine Fehlermeldung.

PRINT queue is empty	Warteschlange ist leer

ERKLÄRUNG:
In der Warteschlange sind keine Dateien mehr gespeichert.

PRINT

Versuchen Sie, eine Datei mit dem Zusatz /T zu löschen, oder wollen Sie Dateien der Warteschlange auf dem Bildschirm anzeigen, erscheint diese Meldung:

PRINT queue is full — Warteschlange ist voll

ERKLÄRUNG:
Sie haben versucht, mehr als die erlaubten 30 Dateien zum Druck zu bestimmen. Dies ist momentan nicht möglich. Geben Sie diese Dateien später ein, wenn schon Dateien ausgedruckt sind.

PROMPT Intern

Zweck

ändert das Systemanforderungszeichen

Syntax

PROMPT [$] [Text]

dabei bedeutet:

$	gefolgt von einem Buchstaben der Tabelle 11.5 ruft es eine besondere Funktion auf
Text	Zeichen, die das neue Prompt bilden

Bemerkungen

Mit diesem Befehl können Sie das *Systemanforderungszeichen* ändern. Es wird auf den von Ihnen eingegebenen Text geändert. Geben Sie vor dem Text ein Dollarzeichen ($) und einen Buchstaben der Tabelle 11.5 ein, rufen Sie eine besondere Funktion auf.

Geben Sie den Befehl ohne weitere Zusatzeingaben ein, erscheint wieder das normale MS-DOS-Prompt (A>).

In der Tabelle 11.5 werden besondere *Funktionen* aufgelistet, die Sie aufrufen durch Eingabe des *Dollarzeichens* mit dem entsprechenden Buchstaben. Die Buchstaben können Sie als Groß- oder Kleinbuchstaben eingeben. Danach können Sie zusätzlich maximal 121 weitere Zeichen als Text eingeben. Sie haben damit die Möglichkeit, Ihr eigenes Prompt zu erzeugen. Allerdings wird dieses Prompt nicht auf Platte gespeichert. Nach jedem Systemstart muß es neu definiert werden (z. B. in der Datei AUTOEXEC.BAT).

PROMPT

Tabelle 11.5 Prompt - Funktionen

ZEICHEN	FUNKTION
$	Dollarzeichen
_	Zeilenvorschub
B	das Zeichen ¦
D	Datum
E	Escape-Sequenz
G	das Zeichen >
H	Schritt zurück
L	das Zeichen <
N	momentanes Laufwerk
P	Name des Inhaltsverzeichnisses
Q	das Zeichen =
S	Leerschritt
T	Uhrzeit
V	Nummer der DOS-Version

Geben Sie z. B. ein

```
PROMPT $N:
```

erzeugen Sie damit das Prompt der MS-DOS-Version 1.25:

```
A:_
```

Wollen Sie jederzeit das aktuelle Verzeichnis in einer eigenen Zeile sehen, geben Sie ein:

```
PROMPT $P$_
```

und das Prompt besteht aus zwei Zeilen. Es könnte z. B. folgendes gezeigt werden:

```
C:\TEXT\BUCH
_
```

PROMPT

Haben Sie die Datei ANSI.SYS als Erweiterung des Betriebssystemes geladen, können Sie mit dem Prompt auch *Escape-Sequenzen* aufrufen, z. B. die Bildschirmsteuerung verändern oder die Tastatur neu belegen. Geben Sie z. B. ein

```
PROMPT $E7m$N:$Em
```

erzeugen Sie das Prompt der Version 1.25. Die Darstellung erfolgt jetzt invers.

PSC Extern

Zweck

gibt Grafikzeichen und Sonderzeichen des Bildschirms auf den Drucker aus.

Syntax

[d:] [Weg] PSCDruckername

dabei bedeutet:

d:	vor dem Programmnamen bestimmt das aufzurufende Laufwerk
Weg	vor dem Programmnamen bestimmt das aufzurufende Verzeichnis
Druckername	nähere Bezeichnung des Druckers: HPTJ - für Hewlwtt Packard Thinkjet Printer IDS - für IDS Prism Drucker OKI - für Okidata Drucker TS315 - für Transtar 315 Color Drucker P920 - für Printek 920 Drucker MX80 - für Epson MX-80 Drucker mit Grafikoption MPI - für MPI Drucker

Bemerkungen

Die Programme PSC können im *Netzwerk* **ohne** Einschränkungen benutzt werden.

Wollen Sie grafische Zeichen des Bildschirmes *ausdrucken*, benötigen Sie zur Unterstützung des Druckers ein Zusatzprogramm. Dieses Zusatzprogramm wird für einige Drucker ebenfalls auf der Systemdiskette 1 mitgeliefert. Da es druckerspezifisch ist, gibt es für jeden Drucker ein eigenes Zusatzprogramm.

PSC

Mit dem Programm CONFIGUR muß der Druckerausgang (PRN) vorher für das jeweilige Gerät angepaßt worden sein. Danach ist es möglich, mit der Tastenkombination **SHIFT-PRTSC** den Bildschirminhalt auf den Drucker zu übertragen.

Voraussetzung ist, daß sich der Bildschirm im Grafikmodus befindet. Das Programm wird folgendermaßen aufgerufen:

PSCDruckername

Für Druckername geben Sie die Abkürzung des entsprechenden Druckers ein. Das Programm wird dann in den Arbeitsspeicher geladen und nicht wieder gelöscht oder überschrieben.

Sie können dann das Arbeitsprogramm aufrufen. Immer wenn Sie jetzt eine Kopie des Bildschirmes benötigen, betätigen Sie die Tasten **SHIFT-PRTSC** gleichzeitig. Der komplette Inhalt des Schirmes wird jetzt inklusive aller Grafik- und Sonderzeichen ausgedruckt.

Ist der Druckvorgang beendet, kann er jederzeit wieder durch die Tastenkombination aufgerufen werden.

Dieses Unterstützungsprogramm ist im Speicher weiterhin vorhanden, selbst wenn Sie den Grafikmodus verlassen. Sie können dann immer noch sämtliche alphanumerischen Zeichen ausdrucken. Kehren Sie in den Grafikmodus zurück, werden wieder sämtliche sichtbaren Zeichen gedruckt.

RDCPM Extern

Zweck

liest CP/M-formatierte Disketten und kopiert CPM-Dateien auf MS-DOS-Disketten.

Syntax

[d:] [Weg] RDCPM
[d:] [Weg] RDCPM ?
[d:] [Weg] RDCPM DIR D: DATEIBEZ /Z
[d:] [Weg] RDCPM S: Ursprungsdatei D: Zieldatei /Z

dabei bedeutet:

d:	vor dem Programmnamen bestimmt das aufzurufende Laufwerk
Weg	vor dem Programmnamen bestimmt das aufzurufende Verzeichnis
?	zeigt das Menü des Programmes
DIR	liest das Verzeichnis der CP/M-Diskette
D:	Bezeichnung des zu lesenden Laufwerkes
Dateibez	genaue Bezeichnung der zu lesenden CP/M-Datei
/Z	Zusatz zum Lesen und Kopieren von Zenith-CP/M-Disketten
S:	Laufwerk mit der zu kopierenden CP/M-Datei
D:	Laufwerk, auf das die CP/M-Datei kopiert werden soll

Bemerkungen

Das Programm RDCPM kann im *Netzwerk* **nicht** benutzt werden.

ACHTUNG:
Benutzen Sie dieses Programm nicht für Laufwerke, denen Sie mit dem Befehl ASSIGN vorübergehend einen anderen Namen zugewiesen haben. Durch die Umbenennung wird nicht auf das ursprüngliche Laufwerk zurückgegriffen, sondern auf das umgeleitete. Dadurch können die wahren Eigenschaften des Laufwerkes verdeckt werden, und das Programm arbeitet nicht fehlerfrei.

Mit diesem Programm können Sie Disketten des *Formates CP/M* lesen und die dort gespeicherten Dateien kopieren. Sie können mit diesem Programm allerdings nur auf CP/M-Disketten zugreifen, nicht auf Winchesterlaufwerke. Haben Sie die CP/M-Dateien auf MS-DOS-Format kopiert (Diskette oder Winchester), können Sie die Daten weiterverarbeiten.

Das Programm RDCPM kann normale IBM-PC CP/M-86 formatierte 5.25-Inch-Disketten mit einer Dichte von 48-tpi (tpi = Tracks per Inch) lesen. Disketten mit einer Dichte von 96-tpi können nicht gelesen werden. Die Disketten können ein- oder beidseitig formatiert sein. Wollen Sie Zenith-formatierte CP/M-Disketten lesen, geben Sie den Zusatz /Z mit ein.

Das Dienstprogramm RDCPM kann drei verschiedene Funktionen ausführen:

- lesen einer CP/M-formatierten Diskette
- kopieren einer CP/M-Datei
- kopieren einer CP/M-Datei und gleichzeitig umbenennen

RDCPM

Rufen Sie das Programm auf durch Eingabe von

RDCPM ?

erscheint auf dem Bildschirm eine Übersicht der möglichen Eingaben:

```
                    RDCPM Version 3.00
Copyright (C) 1985 Zenith Data Systems Corporation

   RDCPM reads a file, file group, or directory from a disk that has been formatted
using a CP/M compatible system and will copy the file(s) to a disk having been
formatted with MS-DOS.

   The RDCPM command is similar to the COPY command, except that the source
file(s) are expected to be on a CP/M disk.

Usage:  RDCPM  source [destination]
-or-    RDCPM  DIR drive

Where "source" is a CP/M filespec, and "destination" a MS-DOS filespec or drive
name.
```

Rufen Sie das Programm ohne Fragezeichen auf, erscheint eine Kurzübersicht:

```
                    RDCPM Version 3.00
Copyright (C) 1985 Zenith Data Systems Corporation

Usage: RDCPM source destination
-or-   RDCPM DIR drive
```

Geben Sie folgenden Befehl ein, lesen Sie das Inhaltsverzeichnis einer CP/M-Diskette im Zenith-Format, die sich im Laufwerk B: befindet:

RDCPM /Z DIR B:

Es könnte folgendes angezeigt werden:

```
              RDCPM Version 3.00
   Copyright (C) 1985 Zenith Data Systems Corporation

LMCMD    CMD : 8087    LIB : DSKCHNG   CMD : CPM      SYS
FLOADER  CMD : PIP     CMD : SUBMIT    CMD : STAT     CMD
TOD      CMD : HELP    CMD : COPYDISK  CMD : DDT86    CMD
ED       CMD : GENCMD CMD : GENDEF     CMD : BLDSYS86 CMD
HDL      CMD :
```

Die Anzeige der Dateien erfolgt immer im CP/M-Format. Wollen Sie nur bestimmte Dateien angezeigt bekommen, können Sie die Dateinamen mit dem Stern (*) oder Fragezeichen (?) zusammenfassen. Ein Beispiel:

RDCPM DIR B:*.CMD

Dateien des Formates CP/M können nur von einer Diskette kopiert werden. Geben Sie den Befehl dazu in folgender Form ein:

RDCPM [S:] Ursprungsdatei [D:] Zieldatei

Der Befehl arbeitet in ähnlicher Weise wie der MS-DOS-Befehl COPY. Zuerst geben Sie die zu kopierende Datei mit dem Namen des Laufwerkes, dem Dateinamen und dem Namenszusatz ein.

Sie können die Datei auf eine MS-DOS-formatierte Diskette oder Winchester kopieren. Zusätzlich können Sie den Namen der Datei ändern. Vor jedem Kopiervorgang wird der Programmname angezeigt:

```
              RDCPM Version 3.00
Copyright (C) 1985 Zenith Data Systems Corporation
```

Es können mit diesem Dienstprogramm folgende CP/M-Formate kopiert werden:

RDCPM

Zenith Data Systems formatiert mit

CP/M-85 SS/DD 48-tpi 5.25 Inch
CP/M-85 DS/DD 48-tpi 5.25 Inch
CP/M-86 SS/DD 48-tpi 5.25 Inch
CP/M-86 DS/DD 48-tpi 5.25 Inch

IBM formatiert mit

CP/M-86 SS/DD 48-tpi 5.25 Inch
CP/M-86 DS/DD 48-tpi 5.25 Inch

Fehlermeldung

Drive not available for CP/M reading	Laufwerk kann vom Programm nicht gelesen werden

ERKLÄRUNG:
Sie haben eine falsche Laufwerkbezeichnung eingegeben, oder Sie haben versucht, Dateien von einer CP/M-formatierten Winchester zu lesen.

HARD DISK ERROR on CP/M drive	Fehler des CP/M-Laufwerkes

ERKLÄRUNG:
Das Programm RDCPM kann die CP/M-Diskette nicht lesen. Die Diskette könnte beschädigt sein.

Source and destination drive must not be the same	Ursprungs- und Ziellaufwerk dürfen nicht gleich sein

ERKLÄRUNG:
Sie haben das Programm RDCPM mit dem gleichen Ziellaufwerk aufgerufen wie das Ursprungslaufwerk. Diese Eingabe kann das Programm nicht bearbeiten. Rufen Sie das Programm mit einem anderen Ziellaufwerk auf.

Insufficient disk space	Nicht genug freie Speicherkapazität vorhanden

ERKLÄRUNG:
Die Zieldiskette kann nicht alle CP/M-Dateien speichern. Löschen Sie einige Dateien auf dem Laufwerk, oder legen Sie eine andere Diskette ein.

No room in directory to create file	Kein Platz mehr im Inhalts-verzeichnis

ERKLÄRUNG:
Sie haben versucht, die CP/M-Dateien in ein Unterverzeichnis einer MS-DOS-Diskette zu kopieren. Dies ist nicht möglich. Sie müssen die Dateien immer in das Rootdir der MS-DOS-Diskette kopieren. Die Meldung erscheint ebenfalls, wenn das Rootdir der Platte voll ist. MS-DOS-Disketten können maximal 112 Einträge im Rootdir speichern. Winchesterteilbereiche können je nach Größe bis zu 512 Einträge speichern.

Source file not found	Datei nicht gefunden

ERKLÄRUNG:
Sie haben das Programm RDCPM mit nicht vorhandenen CP/M-Dateien aufgerufen. Überprüfen Sie Ihre Eingaben.

Cannot read 96 TPI on a 48 TPI drive	96-tpi-Disketten können nicht von einem 48-tpi-Laufwerk gelesen werden

ERKLÄRUNG:
Sie haben in einem 48-tpi-Laufwerk eine 96-tpi-Diskette eingelegt. Das Laufwerk kann die Daten nicht fehlerfrei lesen. Sie müssen erst die Daten mit dem CP/M-Befehl PIP auf eine 48-tpi-Diskette kopieren. Danach können Sie das Programm RDCPM benutzen.

RECOVER Extern

Zweck

erneuert eine Datei oder Diskette mit fehlerhaften Sektoren

Syntax

[d:] [Weg] RECOVER D:
[d:] [Weg] RECOVER Dateibez

dabei bedeutet:

d:	vor dem Programmnamen bestimmt das aufzurufende Laufwerk
Weg	vor dem Programmnamen bestimmt das aufzurufende Verzeichnis
D:	Nach dem Programmnamen bestimmt das Laufwerk mit der fehlerhaften Diskette
Dateibez	Name der Datei mit Laufwerkangabe, das fehlerhafte Sektoren enthält

Bemerkungen

Das Programm RECOVER kann im *Netzwerk* **nicht** benutzt werden.

Mit dem Dienstprogramm RECOVER können Sie Dateien wiederherstellen, die auf Grund fehlerhafter Sektoren nicht mehr einwandfrei arbeiten.

Fehlerhafte Sektoren sind Stellen auf der Plattenoberfläche, die Daten nicht richtig speichern und dadurch eine Fehlermeldung hervorrufen. Diese Fehlermeldung wird ausgegeben, nachdem mehrere Male versucht wurde, auf die Platte zuzugreifen. Nach dieser Fehlermeldung müssen Sie den Vorgang abbrechen. Die Daten, die für diesen Sektor bestimmt waren, sind dann höchstwahrscheinlich verloren.

Werden vom Betriebssystem fehlerhafte Sektoren gemeldet, sollten Sie die Datei, die diesen Fehler hervorruft, mit dem Programm RECOVER überprüfen. Das Programm kann zwar den Fehler nicht korrigieren, aber es kann den Sektor aus der Datei herausnehmen. Sie können dann mit der Datei vielleicht weiterarbeiten oder müssen nur diesen einen Sektor erneuern und nicht die gesamte Datei.

Überprüfen Sie auf einer Diskette immer erst eine einzelne Datei, bevor Sie die gesamte Diskette überprüfen lassen.

Das Dienstprogramm RECOVER überprüft das *FAT* und ordnet danach alle Dateien des Inhaltsverzeichnisses neu. Dabei werden fehlerhafte Sektoren übersprungen. Sie werden im FAT entsprechend gekennzeichnet und können anschließend nicht mehr benutzt werden. Bei diesem Vorgang werden sämtliche Dateinamen überschrieben, unabhängig davon, ob die Datei fehlerhafte Sektoren hat. Da jedes Subdir als einzelne Datei gespeichert wird, ist anschließend die Struktur des Gesamtverzeichnisses verloren. Der neue Dateiname besteht aus einer fortlaufenden vierstelligen Nummer und dem Wort FILE mit dem Namenszusatz .REC. Das Verzeichnis einer erneuerten Platte sieht dann folgendermaßen aus:

```
 Volume in drive C has no label
 Directory of C:\

FILE0001.REC   13312   8-25-84   10:35p
FILE0002.REC   17408   8-25-84   10:35p
FILE0003.REC    1024   8-25-84   10:35p
 .
 .
 .
```

Die laufende *Nummer* der erneuerten Dateien beginnt bei 0001. Als neues Datum und neue Zeit der Dateiänderung wird das momentane Systemdatum eingetragen. Die Größe der Datei ist die neue von RECOVER festgestellte Größe.

Aus diesem Grund sollten Sie immer erst eine einzelne Datei überprüfen.

RECOVER

Überprüfen Sie eine einzelne Datei, rufen Sie das Programm zusammen mit dem kompletten Dateinamen auf, in der Form:

```
RECOVER Dateibez
```

Achten Sie darauf, daß die Programmdiskette in einem anderen Laufwerk ist als die beschädigte Diskette. Sie sollten im Dateinamen keine Globalzeichen verwenden. Das Programm überprüft und erneuert immer nur die erste Datei, deren Dateiname mit dem eingegebenen Namen übereinstimmt. Nachdem Sie die Eingabe mit **RETURN** beendet haben, liest RECOVER jeden Sektor der Datei in den Arbeitsspeicher ein. Der Sektor wird auf Fehler überprüft und wieder auf die Platte zurückgeschrieben. Fehlerhafte Sektoren werden dabei übergangen. Diese Sektoren werden dann auf der Platte markiert, so daß auch später dort keine Daten mehr gespeichert werden können.

Beispiele

Sie haben eine größere Textdatei auf der Diskette im Laufwerk B: mit dem Namen HEFT.DOC. Das Betriebssystem hat wiederholt Zugriffsfehler gemeldet. Sie wollen die Datei überprüfen und fehlerhafte Sektoren löschen. Geben Sie ein:

```
RECOVER B:TEXT.DOC
```

Das Programm antwortet mit der Meldung:

Press any key to begin recovery of the file(s) on drive B	Betätigen Sie eine Taste, damit die Datei im Laufwerk B überprüft werden kann

Das Programm kann jetzt noch mit der Tastenkombination **CTRL-C** abgebrochen werden. Betätigen Sie irgendeine alphanumerische Taste, wird die Datei überprüft. Nach erfolgter Prüfung meldet sich das Programm folgendermaßen:

nnnn of NNNN bytes recovered	nnnn Bytes von NNNN Bytes zurückgeschrieben

Mit der Zahl *nnnn* wird die neue Größe der Datei angegeben. Die Zahl *NNNN* bezeichnet die vorherige Größe. Aus der Differenz ersehen Sie, wieviel Sektoren übersprungen wurden.

Haben Sie von verschiedenen Dateien einer Diskette Fehlermeldungen erhalten, sollten Sie die gesamte Diskette überprüfen. Notieren Sie sich vorher die Dateien der Diskette mit den Größenangaben, um anschließend die erneuerten Dateien mit diesen Daten zu vergleichen. Geben Sie dann ein

RECOVER B:

und es erscheint wieder die Meldung:

Press any key to begin recovery of the file(s) on drive B	Betätigen Sie eine Taste, damit die Datei im Laufwerk B überprüft werden kann

Auch jetzt können Sie das Programm noch mit der Tastenkombination **CTRL-C** abbrechen. Betätigen Sie irgendeine alphanumerische Taste, wird die Diskette überprüft. Nach erfolgter Prüfung meldet sich das Programm folgendermaßen:

nn file(s) recovered	nn Dateien zurückgeschrieben

Waren auf der Diskette mehrere Subdir gespeichert, sind diese jetzt ebenfalls als Dateien vorhanden. Das strukturierte Verzeichnis ist nicht mehr vorhanden. Der Name der Platte wurde gelöscht. Es kann also sein, daß mehr Dateien vorhanden sind als vorher.

Geben Sie den Befehl DIR ein, erscheint das Verzeichnis in der Form:

```
  Volume in drive has no label
  Directory of C:\

FILE0001.REC    13312   8-25-84   10:35p
FILE0002.REC    17408   8-25-84   10:35p
FILE0003.REC     1024   8-25-84   10:35p
 .
 .
 .
FILE000N.REC    25600   8-25-84   10:35p
          nn file(s)   249856 bytes free
```

Durch einen Vergleich mit den Originaldaten des Plattenverzeichnisses können Sie jetzt feststellen, welche Dateien in welchem Umfang geändert wurden. Fehlerhafte Dateien werden kleiner sein, da Sektoren übersprungen wurden. Einige Dateien können auch größer sein als vorher. Dann wurde durch RECOVER der gesamte Sektor der Datei zugeordnet und nicht nur ein Teil, wie es vom Betriebssystem gehandhabt wird.

Fehlermeldungen

File not found — Datei nicht vorhanden

ERKLÄRUNG:
Die bezeichnete Datei ist auf dem angegebenen Laufwerk nicht vorhanden. Überprüfen Sie Ihre Eingabe.

Invalid number of parameters — Falsche Anzahl von Zusatzeingaben

ERKLÄRUNG:
Sie haben den Programmnamen ohne einen Dateinamen oder eine Laufwerkbezeichnung aufgerufen. Überprüfen Sie Ihre Eingabe.

Cannot RECOVER a Network drive	Ein Laufwerk des Netzwerkes kann nicht bearbeitet werden

ERKLÄRUNG:
Sie haben versucht, mit dem Befehl RECOVER ein Laufwerk des Netzwerkes zu bearbeiten. Dies ist nicht möglich.

Invalid DOS Version	Falsche DOS-Version

ERKLÄRUNG:
Sie haben das Programm RECOVER unter einer DOS-Version aufgerufen, die nicht 3.0 oder höher ist.

Warning - directory full	Achtung - Inhaltsverzeichnis voll

ERKLÄRUNG:
Die Meldung erscheint ebenfalls, wenn das Rootdir der Platte voll ist. MS-DOS-Disketten können maximal 112 Einträge im Rootdir speichern. Winchesterteilbereiche können je nach Größe bis zu 512 Einträge speichern.

REM Intern

Zweck

zeigt Kommentare in einer Batchdatei an

Syntax

REM [Text]

Bemerkungen

Dieser Befehl wird hauptsächlich in einer Batchdatei abgearbeitet (s. a. Kapitel 5). Sie können den Befehl aber auch direkt im Betriebssystem eingeben. Schließen Sie jede Zeile eines *Kommentars* mit einem **RETURN** ab.

Sämtliche Eingaben nach dem Befehl werden vom Programm nicht bearbeitet. Sie werden als Kommentar auf dem Bildschirm angezeigt. Erstreckt sich der Kommentar über mehrere Zeilen, muß jede Zeile erneut mit dem Wort REM beginnen. Ebenso ist möglich, zur besseren Lesbarkeit der Batchdatei Leerzeilen einzugeben. Dann besteht jede Zeile nur aus einem Punkt und **RETURN**.

Kommentare werden in Batchdateien eingegeben, um dem Benutzer Bedienungshinweise zu geben. Im folgenden Beispiel formatiert und prüft die Datei neue Disketten. Die Datei hat folgenden Inhalt:

REM Diese Datei formatiert und überprüft neue Disketten.
REM Der Name der Datei ist: NEU.BAT.
PAUSE Legen Sie die neue Diskette in das Laufwerk B:
REM Die neue Diskette wird formatiert.
FORMAT B:
REM Jetzt wird das Verzeichnis der Diskette gezeigt.
DIR B:
REM Jetzt wird die Diskette auf Fehler geprüft.
CHKDSK B:
REM Dies ist das Ende der Datei NEU.BAT.

RENAME oder REN Intern

Zweck

ändert den Namen einer Datei

Syntax

REN Dateibez Dateiname
RENAME Dateibez Dateiname

dabei bedeutet:

Dateibez	Bezeichnung der Datei, die umbenannt werden soll
Dateiname	neuer Name der bezeichneten Datei

Bemerkungen

Mit diesem Befehl *ändern* Sie nur den Namen einer Datei. Die Datei wird nicht kopiert oder auf der Diskette an einen anderen Platz geschrieben. Der Befehl kann mit seinem vollen Namen RENAME oder der Kurzform REN eingegeben werden.

Der erste Dateiname nach dem Befehlsnamen ist immer der Name der umzubenennenden Datei. Diesem Namen kann eine Laufwerkbezeichnung und/oder der Name eines Verzeichnisses vorangestellt werden. Der zweite Name ist der neue Name der Datei. Diesem Namen braucht keine Laufwerkbezeichnung und auch nicht der Name eines Verzeichnisses vorangestellt werden. Entsprechende Eingaben werden ignoriert. Achten Sie darauf, daß der neue Dateiname noch nicht auf der Diskette vergeben ist. In diesem Falle erscheint folgende Fehlermeldung:

Duplicate filename or file not found	Dateiname doppelt, oder Datei nicht vorhanden

Sie können mit dem Befehl mehrere Dateien zugleich umbenennen, wenn Sie im Originalnamen Globalzeichen verwenden.

Benennen Sie **niemals** gekaufte Programme um, es sei denn, Sie werden extra dazu aufgefordert.

Beispiele

Ist im momentanen Verzeichnis die Datei DATEI.ALT gespeichert und soll umbenannt werden, geben Sie ein:

```
REN DATEI.ALT DATEI.NEU
```

Die Datei wird jetzt mit dem Namen DATEI.NEU bezeichnet. Geben Sie den Befehl DIR ein, sehen Sie die Änderung im Verzeichnis.

Die Datei des ersten Beispiels ist nicht im momentanen Verzeichnis gespeichert. Sie ist im Verzeichnis C: FIBU DAT LOHN gespeichert. Um die Datei jetzt umzubenennen, geben Sie ein:

```
REN C: FIBU DAT LOHN DATEI.ALT DATEI.NEU
```

Vor dem ersten Dateinamen (DATEI.ALT) ist die genaue Bezeichnung eingegeben worden. Der zweite Dateiname (DATEI.NEU) wird ohne diese Bezeichnung eingegeben.

Verwenden Sie den Befehl REN mit Globalzeichen (Stern oder Fragezeichen), werden sämtliche erfaßten Dateinamen gleichzeitig umbenannt. Die Globalzeichen müssen bei beiden Namen an gleicher Stelle im Dateinamen stehen. Es werden dann diese Zeichen nicht verändert. Geben Sie ein:

```
REN *.LST *.DOC
```

Sämtliche Dateien mit dem Namenszusatz .LST erhalten jetzt den Namenszusatz .DOC. Der Dateiname wird dabei nicht verändert. Auch hier wäre es möglich, vor dem ersten Namen eine Laufwerkbezeichnung und/oder einen Verzeichnisnamen einzugeben.

Fehlermeldung

Duplicate filename or file not found	Dateiname doppelt, oder Datei nicht vorhanden

ERKLÄRUNG:
Diese Meldung erscheint, bei folgender Falscheingabe:

- der neue Dateiname ist im Verzeichnis bereits vorhanden
- die umzubenennende Datei ist nicht vorhanden
- Sie haben versucht, ein Verzeichnis umzubenennen.

Überprüfen Sie Ihre Eingabe.

RESTORE Extern

Zweck

schreibt die mit BACKUP gesicherten Dateien auf Diskette oder Winchester zurück

Syntax

[d:] [Weg] RESTORE
[d:] [Weg] RESTORE ?
[d:] [Weg] RESTORE [d:] [Dateibez [+Dateibez...] [/X...]

dabei bedeutet:

d: vor dem Programmnamen bestimmt das aufzurufende Laufwerk

Weg vor dem Programmnamen bestimmt das aufzurufende Verzeichnis

d: Name des Laufwerkes mit der Sicherungsdatei

Dateibez Name der gesicherten Datei

/X... Kombination aus folgenden Zusatzeingaben:

/B[:Datum] bedeutet, daß nur Dateien mit einem Herstellungsdatum vor dem eingegebenen Datum zurückgeschrieben werden.

/D[:Datum] bedeutet, daß nur Dateien mit einem Herstellungsdatum nach dem eingegebenen Datum zurückgeschrieben werden.

/E bedeutet, daß die bezeichneten Dateien nicht zurückgeschrieben werden.

/F bedeutet, daß alle Dateien in das momentane Verzeichnis zurückgeschrieben werden.

/P bedeutet, daß Nur-Lese-Dateien abgefragt werden, bevor sie überschrieben werden.

/Q bedeutet, daß jede Datei vor dem Zurückschreiben angezeigt und abgefragt wird.

/R bedeutet, daß mit jeder Meldung ein akustisches Signal erzeugt wird.

/S bedeutet, daß alle Dateien der Platte in allen Verzeichnissen zurückgeschrieben werden.

/T bedeutet, daß alle Dateien mit dem heutigen Datum kopiert werden.

/V bedeutet, daß die kopierten Dateien anschließend auf fehlerhafte Sektoren überprüft werden.

Bemerkungen

Das Dienstprogramm RESTORE schreibt schnell und einfach die mit dem Programm BACKUP gesicherten Dateien von einer Diskette oder Winchester zurück auf das Ursprungslaufwerk. Mit dem Programm können Dateien zwischen allen Peripheriegeräten, mit Ausnahme von CON oder PRN, die ASCII- und Binärzeichen lesen können, geschrieben werden. RESTORE schreibt die Dateien wieder im ursprünglichen Format auf die Platte. Danach kann die Datei normal bearbeitet werden. Das Programm schreibt sowohl alle Dateien als auch nur bestimmte zurück.

Die Sicherungsdatei kann sich über mehrere Disketten erstrecken. Die Disketten wurden dann vom Programm BACKUP durchnumeriert und erhielten damit einen eigenen Namen (Volume). Falls erforderlich, fordert Sie das Programm jeweils auf, die Disketten zu vertauschen.

Programmaufruf

Sie können das Programm RESTORE mit verschiedenen Befehlen aufrufen. Mit Eingabe des Fragezeichens rufen Sie nur das Menü mit allen Befehlserklärungen auf. Geben Sie ein

```
RESTORE ?
```

und es erscheint folgendes:

```
                RESTORE Version 3.04
Copyright (C) 1985 Zenith Data Systems Corporation


The RESTORE utility is designed to complement the BACKUP utility.
RESTORE transfers the specified files from backup disks to their
original drives. RESTORE should only be used on files that were
backed up using the BACKUP utility.

Syntax: RESTORE [<d:> <filespec>[+<filespec>...] [</x>...]]

     Switches: Default state: /Q off and /V off.

/B On or before date. /B:<mm-dd-yy> /Q Query each.
/D On or after date. /D:<mm-dd-yy>  /R Ring bell to prompt.
/E Exception files.                 /S Global subdirectories.
/F Flat restoration.                /T Today's date.
/P Prompt for R/O and written files /V Verify files.

C>
```

Für eine Befehlseingabe ist das Programm erneut aufzurufen. Dabei können Sie zwischen zwei Möglichkeiten wählen:

- Sie geben alle Zusätze in der Befehlszeile ein.
- Sie geben die Zusätze auf eine entsprechende Anfrage ein.

Rufen Sie das Programm ohne weitere Zusätze auf, erscheint der Programmname und das *Prompt* des Programmes ein „Größer-als-Zeichen":

```
RESTORE Version 3.04
Copyright (C) 1985 Zenith Data Systems Corporation
>
```

Sie können jetzt entsprechende Befehle eingeben. Nach jeder Ausführung erscheint wieder das Prompt, und das Programm wartet auf weitere Eingaben. Sie verlassen das Programm erst mit der Eingabe von **RETURN** oder der Tastenkombination **CTRL-C.** Diese Art der Eingabe hat den Vorteil, daß Sie den Programmnamen nicht immer mit eingeben müssen. Die Abarbeitung der Befehle erfolgt genauso, als wären die Eingaben komplett mit dem Befehlsaufruf erfolgt.

Rufen Sie das Programm auf, erscheint folgende Meldung:

```
Insert volume nn in drive d
and press RETURN when ready
```

Haben Sie das Programm gestartet und der Kopiervorgang hat begonnen, zeigt das Programm laufend den jeweiligen Arbeitsgang an. Zwischendurch kann es erforderlich sein, daß Sie die Disketten austauschen müssen. Das Programm RESTORE zeigt zusätzlich das Sicherungsdatum und die laufende Nummer der Diskette. Es könnte folgende Meldung erscheinen:

```
Files were backed up on 11-23-1985

C:Dateiname
C:Dateiname
D:Dateiname
 .
 .
```

Dabei wird dem Dateinamen das Laufwerk vorangestellt, auf das die Datei zurückgeschrieben wird. Dies ist nicht immer das Laufwerk, von dem aus die Datei gesichert wurde.

Erstreckt sich die Sicherungsdatei über mehr als eine Diskette, werden Sie aufgefordert, die Diskette mit der Sicherungsdatei zu wechseln. Es erscheint die Aufforderung:

Insert volume ..., in drive d and press RETURN when ready	Legen Sie die Diskette Nr. ... in das Laufwerk d: und betätigen Sie die Taste RETURN

RESTORE

Findet das Programm RESTORE eine Datei, deren Nur-Lese-Bit gesetzt ist, erscheint die Frage:

The file ... is read-only restore it (Y/N)?	Datei ist schreibgeschützt. Soll trotzdem kopiert werden?

Durch Eingabe von **Y** für Ja, überschreibt das Programm die vorhandene Datei mit der gesicherten Datei. Wollen Sie dies verhindern, geben Sie ein **N**(ein) ein. Die vorhandene Datei bleibt erhalten, und das Programm fährt mit der nächsten Datei fort. Die angefragte Datei wird dann nicht zurückgeschrieben.

Quellaufwerk

Geben Sie hier den Namen des Laufwerkes mit den gesicherten Dateien an. Hier muß eine Diskette enthalten sein mit Dateien, die vom Programm BACKUP erstellt wurden. Eine Angabe ist immer erforderlich. Dies ist die erste Eingabe nach dem Programmnamen RESTORE. Der Name des Quellaufwerkes wird in der Befehlszeile vom Programmnamen durch einen Freiraum getrennt.

Zieldatei (vorherige Ursprungsdatei)

Nach dem Quellaufwerk erfolgt die Eingabe der *Zieldatei*. Geben Sie hier die Dateinamen an, die vorher gesichert wurden. Es muß wenigstens ein Name bezeichnet werden. Der Name der Zieldatei darf dabei die Zeichen zur Zusammenfassung mehrerer Dateien (" * " und " ? ") enthalten.

Es ist auch möglich, mehrere *Zieldateien* einzugeben. Jeder Name darf dann die Zeichen " * " und " ? " enthalten. Die Zieldateien müssen durch ein Pluszeichen (+) miteinander verbunden werden. Andere Zeichen dürfen nicht eingegeben werden. Ebenso dürfen keine Leerräume zwischen den Dateinamen vorhanden sein. Jedem einzelnen Dateinamen darf eine Laufwerkbezeichnung vorangestellt werden.

Normalerweise werden die Dateien vom Programm RESTORE auf das aktuelle Laufwerk zurückgeschrieben. Durch Eingabe eines neuen Ziellaufwerkes können Sie ein anderes Laufwerk bestimmen.

Zusatzeingaben

Über verschiedene *Zusatzeingaben* haben Sie die Möglichkeit, den Programmablauf zu beeinflussen. Alle Eingaben werden zum Schluß der Befehlszeile eingegeben. Sie werden von vorhergehenden Eingaben mit einem Schrägstrich (/) getrennt. Geben Sie keine Zusätze nach dem Namen der Zieldatei ein, führt das Programm folgende Funktionen automatisch aus:

- die Dateien werden mit der alten Verzeichnisstruktur zurückgeschrieben - wird mit dem Zusatz /F geändert
- keine Anfrage vor dem Kopieren jeder Datei - wird mit dem Zusatz /Q geändert.
- keine Fehlerprüfung nach dem Kopieren - wird mit dem Zusatz /V geändert.

/B - vor Datum

Mit diesem Zusatz werden die zurückzuschreibenden Dateien weiter selektiert. Das Programm vergleicht das Datum der Datei im Inhaltsverzeichnis mit dem Datum, daß Sie nach diesem Zusatz eingeben. Die Dateien mit einem *früheren* Datum werden kopiert, andere Dateien werden übergangen. Geben Sie nach dem Zusatz kein Datum ein, ist das momentane Systemdatum das Vergleichsdatum.

Sollen nur die Dateien kopiert werden, deren Herstellungs- oder Änderungsdatum *früher* ist als das momentane (heutige) Datum, geben Sie ein:

```
/B
```

Das Programm vergleicht jetzt das Systemdatum mit dem Datum der ausgewählten Dateien und kopiert nur noch die Dateien, deren Datum *kleiner* ist.

Geben Sie den Zusatz mit einem Datum ein, erfolgt die Datumseingabe in der Schreibweise, die Sie mit dem Befehl COUNTRY bestimmt haben. Datum und Zusatz werden durch einen Doppelpunkt getrennt. Haben Sie das Betriebssystem mit dem Befehl COUNTRY=49 gebootet, erfolgt die Eingabe in der Form:

/B:tt-mm-jj

dabei bedeutet:

tt	Eingabe des Tages mit einer Zahl von 1 bis 31
mm	Eingabe des Monats mit einer Zahl von 1 bis 12
jj	Eingabe des Jahres mit einer Zahl von 00 bis 99

Die einzelnen Zahlen des Datums müssen durch einen Bindestrich getrennt werden. Andere Zeichen sind nicht erlaubt.

Das Programm kopiert dann nur noch die ausgewählten Dateien, deren Datum *kleiner* ist als das eingegebene.

/D - nach Datum

Mit diesem Zusatz werden die zurückzuschreibenden Dateien weiter selektiert. Das Programm vergleicht das Datum der Datei im Inhaltsverzeichnis mit dem Datum, das Sie nach diesem Zusatz eingeben. Die Dateien mit einem späteren Datum werden kopiert, andere Dateien werden übergangen. Geben Sie nach dem Zusatz kein Datum ein, ist das momentane Systemdatum das Vergleichsdatum.

Sollen nur die Dateien kopiert werden, deren Datum der Herstellung oder letzten Änderung später ist als das momentane (heutige) Datum, geben Sie ein:

/D

Das Programm vergleicht jetzt das *Systemdatum* mit dem Datum der ausgewählten Dateien und kopiert nur noch die Dateien, deren Datum größer ist.

Geben Sie den Zusatz mit einem Datum ein, erfolgt die Datumseingabe in der Schreibweise, die Sie mit dem Befehl COUNTRY bestimmt haben. Datum und Zusatz werden durch einen Doppelpunkt getrennt. Haben Sie das Betriebssystem mit dem Befehl COUNTRY=49 gebootet, erfolgt die Eingabe in der Form:

/D:tt-mm-jj

dabei bedeutet:

tt	Eingabe des Tages mit einer Zahl von 1 bis 31
mm	Eingabe des Monats mit einer Zahl von 1 bis 12
jj	Eingabe des Jahres mit einer Zahl von 00 bis 99

Die einzelnen Zahlen des Datum müssen durch einen Bindestrich getrennt werden. Andere Zeichen sind nicht erlaubt.

Das Programm kopiert dann nur noch die ausgewählten Dateien, deren Datum *größer* ist als das eingegebene.

/E - Datei nicht zurückschreiben

Mit diesem Zusatz werden Dateien bezeichnet, die nicht zurückgeschrieben werden sollen. Dies ist hilfreich, wenn Sie mehrere Dateien mit dem Stern oder Fragezeichen zusammenfassen. Alle Dateien, die nach dem Zusatz eingegeben werden, kopiert das Programm nicht. In dieser Eingabe dürfen auch die Zeichen zur Datei-

zusammenfassung verwendet werden. Geben Sie mehrere Dateinamen ein, müssen Sie die Namen durch ein Pluszeichen (+) verbinden. Andere Zeichen dürfen zwischen den Namen nicht eingegeben werden, auch keine Leerzeichen. Die Eingabe erfolgt in der Form:

/E:Dateiname+Dateiname...

Der Zusatz und die Dateinamen werden durch einen Doppelpunkt (:) getrennt. Von den bezeichneten Quelldateien werden jetzt nur noch die Dateien zurückgeschrieben, die durch den Zusatz /E nicht erfaßt wurden. Ein Beispiel:

RESTORE A: /E:*.DAT

Es werden jetzt alle Dateien zurückgeschrieben mit Ausnahme der Dateien, deren Zusatz "DAT" ist.

/F - schreibe in ein Verzeichnis

Normalerweise liest RESTORE alle gespeicherten Informationen über die Ursprungsdateien. Diese Informationen sind Kopf jeder gesicherten Datei gespeichert. Damit kann das Programm alle alten Verzeichnisse wiederherstellen und jede Datei in das entsprechende Verzeichnis zurückschreiben. (Diese Informationen sicherte das Programm BACKUP durch die Eingabe des Zusatzes /G oder /S).

Durch den Zusatz /F weisen Sie das Programm RESTORE an, diese Informationen zu ignorieren und alle Dateien in das momentane Verzeichnis zu schreiben. Damit wird die alte Struktur des Inhaltsverzeichnisses nicht wiederhergestellt.

/P - Nur-Lese-Datei und geänderte Dateien anfragen

Das Programm RESTORE kann normalerweise keine schreibgeschützten Dateien überschreiben. Mit dem Programm ATTRIB können Sie eine Datei schützen, indem Sie das Nur-Lese-Bit setzen. Findet RESTORE auf dem Ziellaufwerk eine Datei mit gesetztem Nur-Lese-Bit und auf dem Ursprungslaufwerk eine gesicherte Datei mit gleichem Namen, erscheint die Frage:

The file is read-only Restore it (Y/N)?	Datei ist schreibgeschützt. Soll trotzdem kopiert werden?

Geben Sie jetzt ein **Y** für Ja ein, wird die vorhandene Datei gelöscht und die gesicherte Datei kopiert. Geben Sie ein N(ein) ein, bleibt die Datei erhalten, und es wird die folgende Datei kopiert.

Mit dem Zusatz /P können Sie ebenfalls verhindern, daß geänderte Dateien überschrieben werden. Das Programm BACKUP markiert jede gesicherte Datei. Das Betriebssystem löscht diese Markierung wieder, und das Programm RESTORE erkennt jetzt, daß die Datei seit der letzten Sichgerung geändert wurde. Findet es gleichzeitig auf der Sicherungsdiskette eine Datei mit diesem Namen, erscheint die Frage:

The file has been modified since it was last backed up. Restore it (Y/N)?

Geben Sie jetzt ein **Y** für Ja ein, wird die vorhandene Datei gelöscht und die gesicherte Datei kopiert. Geben Sie ein **N**(ein) ein, bleibt die Datei erhalten, und es wird die folgende Datei kopiert.

/Q - Datei abfragen

Nach Eingabe dieses Zusatzes sucht das Programm erst alle bezeichneten Quelldateien auf der Sicherungsdiskette. Diese werden dann einzeln angezeigt. Sie werden zu jeder Datei gefragt:

Restore ... (Y/N)?	Soll diese Datei kopiert werden?

Haben Sie die Quelldateien global bezeichnet, ist dieser Zusatz sehr hilfreich. Er zeigt jede Datei einzeln an. Sie können dann entscheiden, ob die Datei kopiert werden soll. Geben Sie ein **Y** für Ja ein, wird die Datei anschließend kopiert. Geben Sie ein **N**(ein) ein, wird die Datei aus der Liste der zu kopierenden Dateien wieder gestrichen.

Sind alle gefundenen Dateien abgefragt, beginnt das Programm mit dem Kopieren.

/R - Summer ertönen lassen

Der Zusatz /R gibt zu jeder Meldung ein akustisches Signal. Dies erfolgt z. B. immer mit der Meldung, die Diskette zu tauschen. Ebenfalls kann damit Ihre Aufmerksamkeit bei einer Fehlermeldung geweckt werden.

/S - kopiere alle Verzeichnisse

Normalerweise kopiert RESTORE nur die bezeichneten Dateien eines Inhaltsverzeichnisses. Alle anderen Dateien, selbst wenn sie mit dem eingegebenen Quelldateinamen übereinstimmen, bleiben unberücksichtigt. Durch Eingabe des Zusatzes /S wird das Programm angewiesen, alle gesicherten Dateien ab dem monentanen Verzeichnis abzusuchen. Geben Sie den Befehl mit dem Rootdir ein, werden alle gesicherten Dateien berücksichtigt. Geben Sie den Befehl mit einem Subdir ein, werden nur die Verzeichnisse ab dem momentanen Verzeichnis abgesucht. Alle höheren Verzeichnisse bleiben weiterhin unberücksichtigt.

/T - heutiges Datum

Mit diesem Zusatz werden die zu kopierenden Dateien weiter selektiert. Das Programm vergleicht das Datum der Datei im Inhaltsverzeichnis mit dem heutigen Datum. Es werden nur noch die bezeichneten Dateien kopiert, deren *Datum* dem Systemdatum entspricht, andere Dateien werden übergangen.

/V - Dateien überprüfen

Mit diesem Zusatz weisen Sie das Programm an, jede Datei sofort nach dem Kopiervorgang mit der Quelldatei zu vergleichen. Das Programm zeigt Ihnen dies durch die Meldung

Verifying ...	Überprüfe Datei

an. Ist die kopierte Datei feherfrei, wird die nächste Datei gesichert. Findet das Programm einen Fehler, erscheint die Frage:

Verify error, try RESTORE again (Y/N)?	Fehler gefunden, soll erneut kopiert werden?

Durch die Eingabe von **Y** für Ja, wiederholen Sie den Kopier- und anschließenden Prüfvorgang. Geben Sie ein **N**(ein) ein, wird mit der nächsten Datei fortgefahren. Die fehlerhafte Datei wird nicht korrigiert.

Beispiele

Die folgenden Beispiele zeigen Ihnen verschiedene Befehlseingaben für das Programm RESTORE. Da das Programm RESTORE das Gegenstück zum Programm BACKUP ist, werden bei den Befehlen die Ursprungs- und Zieldateien vertauscht. Es werden dieselben Beispiele besprochen wie zum Programm BACKUP (s. a. Seite 11.27).

Beispiel 1

Sie haben mehrere Dateien auf der Sicherungsdiskette gespeichert. Die Quelldateien waren nicht alle auf demselben Laufwerk. Sie hatten die Dateien gesichert mit dem Befehl:

```
BACKUP A:*.BAS+B:KAP.*+C:???X.COM F: /R /N
```

Sie schreiben die Dateien zurück mit dem Befehl:

```
RESTORE F: A:*.*+B*.*+C:*.* /R /F
```

Mit dem Backup-Befehl wurde bereits eine Auswahl der Dateien getroffen. Sie können daher im Restore-Befehl mit dem Stern alle Dateinamen global erfassen. Durch den Zusatz /R bestimmen Sie, daß alle Dateien vorher angezeigt werden. Der Zusatz /F schreibt alle Dateien in das aktuelle Inhaltsverzeichnis, unabhängig vom alten Originalverzeichnis.

Beispiel 2

Sie haben vom Laufwerk F: Dateien mit den Zusätzen .LTR, und .TXT gespeichert. Die ebenfalls vorhandenen Dateien mit dem Zusatz .DAT waren nicht gesichert worden. Es waren die Dateien aus allen Verzeichnissen gesichert worden. Sie hatten folgenden BACKUP-Befehl eingegeben:

```
BACKUP F:*.* A: /G /E:*.DAT
```

RESTORE

Sie schreiben die Dateien zurück mit dem Befehl:

RESTORE A: F:*.TXT /F /V

Jetzt werden alle Dateien der Sicherungsdatei mit dem Zusatz .TXT auf das alte Laufwerk zurückgeschrieben. Die Dateien werden aber nicht wieder in das strukturierte Inhaltsverzeichnis geschrieben, das mitgespeichert ist, sondern alle in ein Verzeichnis. Jede kopierte Datei wird anschließend auf Fehler überprüft (s. a. Zusatz /V).

Beispiel 3

Die Eingabe des letzten Beispiels läßt sich leicht variieren durch den Zusatz /Q. Jetzt werden sämtliche bezeichneten Dateien vorher abgefragt, und Sie haben die Möglichkeit, einzelne Dateien zu löschen. Geben Sie ein:

RESTORE A: F:*.TXT /F /V /Q

Es erscheint jetzt nach einem kurzen Moment für jede Datei die Frage:

RESTORE F:... (Y/N)? Soll diese Datei kopiert werden?

Geben Sie ein **Y** für Ja ein, wird die Datei anschließend gesichert. Geben Sie ein **N**(ein) ein, wird die Datei aus der Liste gestrichen.

Beispiel 4

Sie können das Programm RESTORE auch ohne weitere Zusätze in der Befehlszeile aufrufen. Sie geben dann die entsprechenden Befehle nacheinander ein. Rufen Sie das Programm auf

RESTORE

es erscheint der Programmname und das Prompt (>):

```
                    RESTORE Version 3.04
Copyright (C) 1985 Zenith Data Systems Corporation
>
```

Sie können jetzt mehrere Befehle eingeben, ohne immer wieder das Programm neu aufrufen zu müssen.

Sie können jetzt alle Dateien mit dem Zusatz .TXT auf ein Laufwerk zurückschreiben, auf dem eventuell schon vorhandene Dateien gleichen Namens überschrieben werden sollen. Geben Sie ein:

```
RESTORE A: *.* /Q /V
```

Das Programm verlassen Sie mit der Eingabe von **RETURN** nach dem RESTORE-Prompt (>). Es erscheint wieder das DOS-Prompt (C>).

Kopieren mit einer Batchdatei

Das Programm übergibt bei seinem Ende dem Betriebssystem immer einen Wert. Mit dem Befehl ERRORLEVEL können Sie diesen Wert in einer Batchdatei abfragen und den weiteren Programmablauf steuern. Die folgende Tabelle zeigt eine Übersicht.

Tabelle 11.6 Ausgangswerte des Programmes RESTORE

WERT	BEDEUTUNG
0	Programm fehlerfrei beendet
1	keine Dateien gefunden
2	Fehler durch Netzwerk
3	vom Bediener abgebrochen
4	durch einen Fehler abgebrochen

RESTORE

Fehlermeldungen

File ... already exists, do you wish to delete it (Y/N)?	Datei ... ist schon vorhanden, soll die Datei gelöscht werden?

ERKLÄRUNG:
Das Programm hat auf der Zieldiskette eine Datei gefunden, die jetzt mit dem Programm RESTORE auf die Platte zurückgeschrieben werden soll. Da Sie den Zusatz /O nicht eingegeben haben, wird die Datei nicht automatisch überschrieben. Soll die Datei überschrieben werden, geben Sie ein **Y** für Ja ein. Soll die Datei nicht überschrieben werden, geben Sie ein **N**(ein) ein. Die Datei auf der Zieldiskette bleibt dann erhalten. Die Datei der Sicherungsdiskette wird nicht auf die Zieldiskette geschrieben.

File ... not found	Datei ... nicht gefunden

ERKLÄRUNG:
Die bezeichnete Datei ist auf der Sicherungsdiskette nicht vorhanden. Überprüfen Sie Ihre Eingabe auf einen Schreibfehler. Sie können auch das interne Verzeichnis der Sicherungsdiskette nach der bezeichneten Datei absuchen.

Invalid backup file	Fehlerhafte Datei

ERKLÄRUNG:
Die von Ihnen benannte Datei ist fehlerhaft. Es ist auch möglich, daß Sie eine gesicherte Datei mit dem Zusatz .000 benannt haben. Überprüfen Sie Ihre Eingabe und ggf. die Diskette mit CHKDSK.

Invalid date in switch	Falsche Datumseingabe

ERKLÄRUNG:
Das Datum, das Sie mit dem Zusatz /B oder /D eingegeben haben, ist falsch. Überprüfen Sie Ihre Eingabe.

Invalid drive designation on RESTORE file	Falsche Laufwerkbezeichnung

EDRKLÄRUNG:
Sie haben eine falsche Laufwerkbezeichnung eingegeben. Das Laufwerk ist nicht vorhanden oder wird vom Programm nicht erkannt. Es können die Laufwerke A: bis H: benannt werden.

Invalid exception file specifications	Falsche Dateibezeichnung für die mit /E bezeichneten Dateien

ERKLÄRUNG:
Sie haben den Zusatz /E ohne Dateiname oder den Dateinamen falsch eingegeben. Überprüfen Sie Ihre Eingabe.

Invalid file name	Falscher Dateiname

ERKLÄRUNG:
Der von Ihnen benannte Dateiname enthält ungültige Zeichen. Überprüfen Sie Ihre Eingabe.

Invalid selection file specifications	Falsche Eingabe der Dateiauswahl

ERKLÄRUNG:
Sie haben Dateinamen oder Zusätze falsch eingegeben. Überprüfen Sie Ihre Eingabe.

Invalid switch /X specified	Falsche Zusatzeingabe /X benannt

ERKLÄRUNG:
Sie haben nach dem Schrägstrich (/), der den folgenden Buchstaben als Zusatzeingabe definiert, ein falsches Zeichen eingegeben. Überprüfen Sie Ihre Eingabe.

RESTORE

No files selected	Keine Dateien benannt

ERKLÄRUNG:
Sie haben in der Befehlszeile keine Dateinamen benannt, oder die eingegebenen Namen erfüllen nicht die Bedingungen der Zusatzeingaben. Diese Meldung erscheint auch, wenn Sie auf eine Frage des Zusatzes /Q mit **N**(ein) antworten.

Not enough parameters specified	Nicht alle Eingaben vorgenommen

ERKLÄRUNG:
Sie haben die Befehlszeile unvollständig abgeschlossen. Überprüfen Sie Ihre Eingaben.

Out of disk on restoration of .... insert another disk and press RETURN, or press any other key to abort	Speicherkapazität der Platte ist für die Datei ... erschöpft. Legen Sie eine andere Diskette ein und betätigen Sie RETURN, oder jede andere Taste zum Abbruch

ERKLÄRUNG:
Die gerade kopierte Datei kann nicht komplett auf die Zieldiskette geschrieben werden. Die Diskette ist voll. Legen Sie eine neue, formatierte Diskette in das Laufwerk, und betätigen Sie die Taste **RETURN.**

Too many parameters specified	Zu viele Zusatzeingaben

ERKLÄRUNG:
Sie haben mehr Zusätze eingegeben, als vom Programm verarbeitet werden können. Überprüfen Sie Ihre Eingabe.

Verify error, try restore again (Y/N)?	Prüffehler, soll die Datei erneut kopiert werden (J/N)?

ERKLÄRUNG:
Die zu kopierende Datei wurde nicht korrekt auf die Diskette geschrieben. Geben Sie ein **Y** für Ja ein, wird die Datei erneut kopiert. Geben Sie ein **N**(ein) ein, wird mit der nächsten Datei fortgefahren.

RMDIR oder RD **Intern**

Zweck

löscht ein Inhaltsverzeichnis aus der Gesamtstruktur

Syntax

RMDIR [D:] Weg
RD [D:] Weg

dabei bedeutet:

D: Bezeichnung des Laufwerks, auf dem das Verzeichnis gelöscht werden soll.

Weg mehrere Inhaltsverzeichnisse in Folge in der Form:

[\] [Verzeichnis [\Verzeichnis ...]]

Bemerkungen

Mit diesem Befehl wird ein Inhaltsverzeichnis *komplett gelöscht.* Voraussetzung dafür ist, daß im Inhaltsverzeichnis keine Dateien mehr gespeichert sind. Es dürfen nur noch die beiden DOS-Einträge (. und ..) vorhanden sein. Wollen Sie z. B. vom momentanen Verzeichnis aus das Verzeichnis TEXT löschen, geben Sie ein:

RMDIR TEXT

Sie können den Befehl auch in der Kurzform eingeben:

RD TEXT

Mit diesem Befehl können Sie keine Dateien löschen. Versuchen Sie ein Verzeichnis zu löschen, in dem noch Dateien gespeichert sind, erscheint die Fehlermeldung:

Invalid path, not directory, or directory not empty	Falscher Weg, kein Verzeichnis, oder Verzeichnis nicht leer

RMDIR oder RD

Sie können mit diesem Befehl auch nicht das momentane Verzeichnis löschen, ebenso keine Verzeichnisse, die in der Gesamtstruktur höher als das momentane Verzeichnis sind.

Sie können aber jedes beliebige Verzeichnis löschen, das in der Struktur unter dem momentanen Verzeichnis angelegt wurde und keine Dateien mehr gespeichert hat. Sind in dem zu löschenden Verzeichnis noch Dateien gespeichert, können Sie diese mit dem Befehl DIR löschen (s. a. Erklärung des Befehls DIR).

Soll z. B. ein Verzeichnis gelöscht werden, das mehrere Verzeichnisse tiefer als das momentane Verzeichnis ist, geben Sie nach dem Befehlsnamen alle Verzeichnisse ein, das zu löschende Verzeichnis als letztes. Ein Beispiel:

```
RD TEXT\SPEICHER\TEST\TEST1
```

Das Verzeichnis TEST1 wird jetzt gelöscht.

Ebenso ist es möglich, ein Verzeichnis zu löschen, das auf einem anderen Ast des Verzeichnisses angelegt wurde. Dafür ist es wieder erforderlich, daß Sie alle Verzeichnisse auf dem Weg dahin benennen. Ein Beispiel:

```
RD ..\..\BASIC\SPEICHER\DATEI
```

Sie gehen vom momentanen Verzeichnis zweimal zurück. Dies wird durch die beiden Punkteingaben (..) angegeben. Damit sind Sie im Rootdir. Jetzt gehen Sie durch Eingabe der Verzeichnisnamen über das Verzeichnis BASIC zum Verzeichnis SPEICHER und löschen dort das Verzeichnis DATEI.

Fehlermeldungen

Invalid number of parameters — Falsche Anzahl von Zusatzeingaben

ERKLÄRUNG:
Sie haben den Befehl ohne weitere Angaben eingegeben. Sie müssen mindestens den Namen eines Verzeichnisses eingeben.

Invalid path, not directory, or directory not empty	Falscher Weg, kein Verzeichnis, oder Verzeichnis nicht leer

ERKLÄRUNG:
Sie haben einen falschen Namen für das Verzeichnis angegeben, oder im bezeichneten Verzeichnis sind noch Dateien gespeichert, oder Sie haben versucht ein Verzeichnis zu löschen, das nicht unterhalb des momentanen Verzeichnisses ist.

SEARCH Extern

Zweck

sucht Dateien im hierarchischen Inhaltsverzeichnis der Platte

Syntax

[d:] [Weg] SEARCH zDn /X

dabei bedeutet:

d: vor dem Programmnamen bestimmt das aufzurufende Laufwerk

Weg vor dem Programmnamen bestimmt das aufzurufende Verzeichnis

zDn Namen der zu suchenden Dateien. Die Namen können sowohl mit den Zeichen „*“ und „ ? “ zusammengefaßt werden, als auch mit vorangestelltem Laufwerk und/oder Verzeichnis eingegeben werden.

/X Kombination aus folgenden Zusatzeingaben:

/C veranlaßt das Programm, keine Subdirectories abzusuchen.

/D veranlaßt das Programm, die gerade durchsuchten Verzeichnisse anzuzeigen.

/T veranlaßt das Programm, die Datei- und Verzeichnisnamen versetzt anzuzeigen. Dadurch wird die Struktur des gesamten Verzeichnisses sichtbar.

Bemerkungen

Das Programm SEARCH kann im *Netzwerk* **ohne** Einschränkungen benutzt werden.

Das Programm *sucht* die bezeichneten Dateien in der hierarchischen Struktur des Gesamtverzeichnisses, beginnend ab dem momentanen Verzeichnis. Es zeigt das genaue Verzeichnis der Datei mit seinem kompletten Namen an. Durch Eingabe des Zusatzes /C weisen Sie das Programm an, nur das momentane Verzeichnis abzusuchen. Soll die gesamte Platte abgesucht werden, muß das Programm vom Rootdir aus gestartet werden. Die Ausgabe der Namen erfolgt auf dem Standardausgabegerät (Bildschirm). Dies kann für die Befehlsverkettung (z. B. mit APPLY) sehr nützlich sein.

Sie sind im Rootdir und suchen sämtliche Batchdateien der Platte. Die Ausgabe soll strukturiert erfolgen. Geben Sie ein:

```
SEARCH *.BAT /T
```

Jetzt werden sämtliche Batchdateien mit ihrem kompletten Namen und Verzeichnis angezeigt.

Wollen Sie gleichzeitig auch noch den Inhalt jeder Batchdatei sehen, brauchen Sie die Ausgabe des Programmes SEARCH nur als Eingabe des Programmes APPLY nehmen. Dies führt dann für jede Datei den Befehl TYPE aus. Geben Sie ein:

```
SEARCH *.BAT ¦APPLY "TYPE %"
```

SELECT Extern

Zweck

wählt Tastaturanpassung und Datumsformat aus

Syntax

[d:] [Weg] SELECT yyy xxxx

dabei bedeutet:

d:	vor dem Programmnamen bestimmt das aufzurufende Laufwerk
Weg	vor dem Programmnamen bestimmt das aufzurufende Verzeichnis
yyy	Landeskennzifer. Damit wird das Format von Datum und Uhrzeit bestimmt.
xxxx	Landesname. Damit wird das Tastaturanpassungsprogramm bestimmt.

Bemerkungen

Das Programm SELECT kann im *Netzwerk* **nicht** benutzt werden.

Ihre MS-DOS-*Systemdisketten* können Sie entweder mit dem Befehl DISKCOPY oder dem Befehl SELECT kopieren. Der Befehl SELECT erstellt gleichzeitig zwei Dateien auf der neuen Diskette, die das Betriebssystem während des Bootens abarbeitet. Damit werden Datum und Zeit im gewohnten Format dargestellt und die Tastatur entspricht der deutschen DIN-Norm. Das Programm DISKCOPY wird zusätzlich benötigt.

Benutzen Sie das Programm SELECT nur zum Kopieren von Systemdisketten.

Legen Sie in das Laufwerk A: die original Systemdiskette I (Schreibschutz nicht vergessen!) und in das Laufwerk B: eine neue noch nicht benutzte Diskette. Wählen Sie die Werte für die Landeskennziffer und den Namen aus der Tabelle 11.7. Für ein deutsches Betriebssystem rufen Sie das Programm SELECT mit folgendem Befehl auf:

```
SELECT 049 GR
```

Das *Tastaturprogramm* hat den kompletten Namen KEYBGR.COM und sollte immer benutzt werden. Die 7-Bit-Fassung mit dem Namen KEYBAGR.COM benötigen Sie für einen 7-Bit-Drucker, der deutsche Sonderzeichen an Stelle der amerikanischen Sonderzeichen darstellen kann. Benutzen Sie das Programm KEYBAGR.COM, werden nicht mehr von allen Programmen deutsche Sonderzeichen auf dem Bildschirm dargestellt. Dies gilt immer dann, wenn diese Programme Zeichen direkt darstellen und das Betriebssystem nicht benutzen.

Tabelle 11.7 Zusatzeingaben für SELECT

LAND	KENN- ZIFFER	Tastatur 7-BIT	 8-BIT
Australien	061	AUS	US
Belgien	032	AFR	FR
Dänemark	045	ADA	DA
Deutschland	049	AGR	GR
Finnland	358	ASW	SW
Frankreich	033	AFR	FR
Großbritanien, UK	044	AUK	UK
Großbritanien, US	044	AUS	US
Israel	972	---	---
Italien	039	AIT	IT
Kanada, EL	001	---	CANE
Kanada, SE	001	---	CANS
Niederlande	031	AUS	US
Norwegen	047	ANO	NO
Schweiz, DT	041	---	CHG
Schweiz, FR	041	---	CHF
Spanien	034	ASP	SP
USA	001	AUS	US

Fehlermeldungen

Cannot SELECT a Network drive	Ein Laufwerk des Netzwerkes kann nicht kopiert werden

ERKLÄRUNG:
Sie haben versucht, mit dem Befehl SELECT ein Laufwerk des Netzwerkes zu kopieren. Dies ist nicht möglich.

Cannot execute DISKCOPY	Das Programm DISKCOPY kann nicht aufgerufen werden

ERKLÄRUNG:
Das Programm SELECT kann das Programm DISKCOPY nicht finden. Legen Sie eine Diskette mit beiden Programmen in das Laufwerk, oder geben Sie für DISKCOPY ein Verzeichnis mit dem Befehl PATH an.

Invalid country code	Falsche Landeskennziffer
Invalid keyboard code	Falscher Tastaturname

ERKLÄRUNG:
Sie haben als Zusatzeingabe für SELECT falsche Werte eingegeben. Erlaubt sind nur Werte der Tabelle 11.7. Rufen Sie das Programm neu auf.

Missing country code	Landeskennziffer fehlt
Missing keyboard code	Tastaturname fehlt

ERKLÄRUNG:
Sie haben das Programm SELECT ohne Zusatzeingaben aufgerufen. Rufen Sie das Programm mit einem der Werte aus Tabelle 11.7 erneut auf.

SET Intern

Zweck

setzt den Wert einer Zeichenkette gleich dem Wert einer anderen Zeichenkette

Syntax

SET
SET Zeichenkette1=Zeichenkette2

dabei bedeutet:

Zeichen-kette1	zu definierender Wert
Zeichen-kette2	gespeicherter Wert für Zeichenkette1

Bemerkungen

Mit diesem Befehl *definieren* Sie Werte, die von verschiedenen Programmen abgefragt und danach entsprechend verarbeitet werden können. Voraussetzung ist immer, daß das Programm diese Variablen abfragt.

In der Datei COMMAND.COM wird von MS-DOS ein *Zwischenspeicher* mit einer Kapazität von 200 Zeichen eingerichtet. Dort werden alle Variablen gespeichert, auf die über das Betriebssystem zurückgegriffen wird. Durch Eingabe des Befehls ohne Zusätze, wird der gesamte Speicherinhalt angezeigt. Geben Sie ein

SET

und es könnte folgendes angezeigt werden:

COMSPEC=C:\COMMAND.COM
PATH=

SET

Nach dem Systemstart ist nur der Dateiname des Befehlsinterpreters dort gespeichert. Das ist in diesem Falle die Datei COMMAND.COM auf dem Laufwerk C:, da in der Datei CONFIG.SYS kein anderer Befehl vorhanden war (s. a. Kapitel 9). Weitere Variable, die Sie hier speichern, sind die Werte für PATH oder PROMPT.

Haben Sie eine *Variable* definiert und in diesem Zwischenspeicher abgelegt, bleibt sie dort solange erhalten, bis sie neu definiert oder wieder gelöscht wird. Die Variablen müssen nur nach einem Systemstart definiert werden.

Das Betriebssystem und jedes Anwenderprogramm kann diesen Speicher abfragen und eine oder alle Variable verarbeiten. Sie haben die Möglichkeit, fast jeden Wert zu speichern. Einzige Voraussetzung, Sie geben den Befehl in der richtigen Schreibweise ein.

Nach dem Befehlsnamen wird der Name der Variablen eingegeben. Dieser Name wird von MS-DOS immer in Großbuchstaben gespeichert. Auf die Variable folgt ohne Leertaste ein Gleichheitszeichen. Sämtliche Zeichen danach werden als Wert der Variable gespeichert. Diese Zeichen werden exakt so gespeichert, wie sie eingegeben werden. Es werden also z. B. Kleinbuchstaben nicht mehr umgeformt. Ein Beispiel:

```
SET VAR=Beispiel Datei
```

Der Variablen VAR ist jetzt die Zeichenfolge „Beispiel Datei" zugeordnet. Geben Sie den Befehl SET noch einmal ein, erscheint jetzt folgendes:

```
COMSPEC=C:\COMMAND.COM
PATH=
VAR=Beispiel Datei
```

Diesen Wert können Sie jetzt z. B. in der Batchdatei mit dem Namen TEST.BAT verarbeiten. Die Datei hat folgenden Inhalt:

```
REM Dies ist eine Beispieldatei
ECHO OFF
ECHO In der folgenden Zeile wird die Variable VAR gezeigt:
ECHO %VAR%
PAUSE
 ...
 ...
```

Die *Variable* (VAR) muß in der Batchdatei immer von Prozentzeichen eingeschlossen werden. Daran erkennt das System, daß es sich um eine Variable handelt. Rufen Sie die Datei auf, erscheinen auf dem Bildschirm folgende Zeilen:

```
REM Dies ist eine Beispieldatei
ECHO OFF
In der folgenden Zeile wird die Variable VAR gezeigt:
Beispiel Datei
Strike a key when ready ...
```

Mit der Betätigung einer alphanumerischen Taste wird die Datei weiter abgearbeitet. Mit der Kombination **CTRL-C** verlassen Sie die Datei wieder.

Sie löschen die Variable wieder durch Eingabe von

```
SET VAR=
```

Soll die Variable neu definiert werden, wird der alte Wert überschrieben. Es ist nicht erforderlich, ihn erst zu löschen.

Mit dem Befehl SET lassen sich auch für bestimmte Anwenderprogramme die Namen der Gerätetreiber ändern. Ein Beispiel:

```
SET TTY=VT52
```

ändert den Namen des Konsoltreibers von TTY auf VT52.

Fehlermeldung

Out of environment space	Kapazität des Speichers erschöpft

ERKLÄRUNG:
Der Speicher von 200 Bytes ist voll. Überprüfen Sie mit Eingabe von SET den Inhalt, und löschen Sie nicht mehr benötigte Werte.

SHARE — Extern

Zweck

reserviert Platz für File-Sharing

Syntax

[d:] [Weg] SHARE [/F:n] [/L:n]

dabei bedeutet:

d:	vor dem Programmnamen bestimmt das aufzurufende Laufwerk
Weg	vor dem Programmnamen bestimmt das aufzurufende Verzeichnis
/F:n	reserviert Platz für Dateiinformationen. Ohne Eingabe des Zusatzes ist Platz für ca. 90 Dateien vorhanden.
/L:n	bestimmt die maximale Zahl der Dateisperrungen im Netzwerk. Ohne Eingabe des Zusatzes können maximal 20 Sperrungen gleichzeitig bearbeitet werden.

Bemerkungen

Arbeiten Sie im *Netzwerk,* können Sie Dateien auf einem anderen Computer bearbeiten. Dies kann z. B. auch gleichzeitig mit einem anderen Benutzer zusammen geschehen. Dafür benötigt das Betriebssystem zusätzliche Informationen über die gerade bearbeitete Datei. Das Programm SHARE reserviert Speicherplatz für diese Informationen. Für jede Datei, die im Netzwerk bearbeitet werden soll, werden 11 Bytes für allgemeine Informationen reserviert und maximal weitere 11 Bytes für den Dateinamen (8 Bytes für den Namen, 3 Bytes für den Zusatz). Besteht der Dateiname aus weniger Zeichen, benötigt er entsprechend weniger Speicherplatz.

Das Programm SHARE reserviert weiteren *Speicherplatz,* damit das Betriebssystem ebenfalls Informationen über Dateisperrungen speichern kann.

Nachdem Sie das Programm SHARE aufgerufen haben, prüft MS-DOS jeden Dateizugriff mit Hilfe der hier gespeicherten Informationen. Ist ein Zugriff nicht erlaubt, wird eine entsprechende Fehlermeldung ausgegeben.

Das Programm kann nur einmal aufgerufen werden. Dies sollte direkt nach dem Systemstart mit Hilfe der Datei AUTOEXEC.BAT geschehen. Weitere Informationen über Netzwerk finden Sie im Handbuch für Programmierer.

Programmaufruf

Rufen Sie das Programm mit seinem Namen auf. Geben Sie keinen Zusatz ein, wird Platz für ca. 90 Dateien und ca. 20 Dateisperrungen reserviert. Geben Sie ein:

```
SHARE
```

Reicht der reservierte Platz nicht aus, können Sie mit Hilfe der Zusatzeingaben mehr Platz reservieren.

/F:n -- Dateiinformationen

Jede Datei, die von einem Programm im Netzwerkmodus geöffnet wird, benötigt Speicherplatz für zusätzliche Informationen. Das Betriebssystem kontrolliert damit alle Zugriffe. Dafür speichert MS-DOS einmal den vollen Dateinamen (maximal 11 Zeichen) und in weiteren 11 Bytes zusätzliche Informationen. Sie müssen daher für jede Datei den Speicherplatz errechnen (1 Byte je Zeichen des Dateinamens plus 11 weitere Bytes). Geben Sie den errechneten Wert mit dem Zusatz /F ein. Ohne diesen Zusatz reserviert das Programm 2048 Bytes.

/L:n -- Dateisperrungen

Jeder Bediener, der eine Datei im Netzwerkmodus öffnet, kann gleichzeitig bestimmen, ob ein weiterer Bediener mit der Datei arbeiten darf. Das Anwenderprogramm teilt dem Betriebssystem mit, ob Teile der Datei oder die gesamte Datei für weitere Bediener vorrübergehend gesperrt werden. Auch zur Speicherung dieser Informationen benötigt MS-DOS zusätzlichen Speicherplatz. Diesen reservieren Sie mit dem Zusatz /L. Ohne Eingabe des Zusatzes können maximal Informationen für 20 Sperrungen gespeichert werden.

Fehlermeldungen

Incorrect Parameter	Falsche Zusatzeingabe

ERKLÄRUNG:
Sie haben den Zusatz /F oder /L mit einem falschen Wert aufgerufen. Überprüfen Sie Ihre Eingabe.

SHARE already installed	Das Programm SHARE wurde bereits aufgerufen

ERKLÄRUNG:
Sie haben das Programm SHARE bereits aufgerufen. Sollen andere Werte gespeichert werden, müssen Sie das System neu starten.

Insufficient memory	Arbeitsspeicher zu klein

ERKLÄRUNG:
Der freie Arbeitsspeicher ist zu klein, um den gewünschten Speicher zu reservieren. Starten Sie das System neu, und laden Sie weniger residente Programme. Bestimmen Sie mit dem Befehl BUFFERS weniger Zwischenspeicher für Plattenzugriffe.

SHIFT Intern

Zweck

erhöht die Zahl der Variablen in einer Batchdatei auf mehr als 10

Syntax

SHIFT

Bemerkungen

Damit eine einmal geschriebene Batchdatei flexibel eingesetzt werden kann, ist es möglich, *Variable* zu verarbeiten. Es kann eine beliebige Anzahl von Variablen eingesetzt werden. Die Variablen werden in der Datei mit den Ziffern 0 bis 9 (gleich 10 Variable) und einem vorangestellten Prozentzeichen (%) bezeichnet. Die entsprechenden Werte dafür werden in der Befehlszeile hinter dem Dateinamen eingegeben. Jeder Wert wird durch ein Freizeichen getrennt. Dabei werden die eingegebenen Werte in ihrer Reihenfolge in der Datei ersetzt. Der erste zu ersetzende Wert ist der Dateiname selber (%0). Dadurch ist es möglich, daß sich die Datei selber wieder aufruft. Ein Beispiel:

```
Dateiname   Variable 1   Variable 2   Variable 3...   Variable 9
!           !            !            !               !
!           !            !            !               !
%0          %1           %2           %3...           %9
```

Mit dem Befehl SHIFT ist es möglich, diese *Begrenzung* aufzuheben. Jeder Aufruf von SHIFT in der Datei verschiebt die Variablen um eine Stelle nach links. Damit wird aus der 11. Variablen die 10. oder aus der 3. Variablen die 2. Die folgende Gegenüberstellung soll dies verdeutlichen:

Variable vor Aufruf von SHIFT:	Variable nach Aufruf von SHIFT:
%0 TEST	%0 Datei1
%1 Datei1	%1 Datei2
%2 Datei2	%2 Datei3
%3 Datei3	%3 Datei4
...	...
...	...
...	...
%9 Datei9	%9 Datei10

Befehlserklärungen

Befehlserklärungen

SHIFT

Der Befehl SHIFT wird in der Batchdatei ohne weitere Zusätze eingegeben. Damit er sinnvoll eingesetzt werden kann, sollte er mit dem Befehl GOTO kombiniert werden. Geben Sie z. B. folgende Batchdatei ein:

```
REM Dies ist die Datei STREICHE.BAT
:ANFANG
IF "%1" == "" GOTO ENDE
DEL %1
SHIFT
GOTO ANFANG
:ENDE
```

Mit diesem Beispiel können Sie mehrere Dateien gleichzeitig löschen. Rufen Sie die Datei folgendermaßen auf:

```
STREICHE DATEI1 DATEI2 DATEI3 DATEI4
```

Die Datei wird in folgenden Schritten abgearbeitet:

Schritt 1 Es wird der Kommentar angezeigt.

Schritt 2 Durch den Doppelpunkt wird die Zeile als Bezeichnung für den Befehl GOTO (s. a. Seite 11.167) gekennzeichnet.

Schritt 3 Jetzt wird die Variable überprüft. Ist ihr Wert gleich null (" "), wird der zweite Teil des Befehls abgearbeitet (GOTO ENDE).

Zu Beginn wird der erste Dateiname, in diesem Beispiel DATEI1, für die Variable (%1) eingesetzt.

Schritt 4 Die Datei wird gelöscht.

Schritt 5 Mit dem Befehl SHIFT werden alle Variablen der Befehlszeile um eine Stelle nach links gesetzt. Die DATEI2 wird jetzt vom Programm als Variable %1 erkannt.

Schritt 6 Das Programm springt zurück zur Zeile 2 (:ANFANG)

Schritt 7 Ab dem Schritt 2 wird das Programm fortgesetzt.

Diese Schleife wird jetzt so oft durchlaufen, wie die Bedingung in Zeile 3 erfüllt wird. Dann verzweigt das Programm zur Bezeichnung ENDE und ist beendet.

SORT — Extern

Zweck

liest beliebige Daten und gibt sie sortiert wieder aus

Syntax

[d:] [Weg] SORT [/X]

dabei bedeutet:

d:	vor dem Programmnamen bestimmt das aufzurufende Laufwerk
Weg	vor dem Programmnamen bestimmt das aufzurufende Verzeichnis
/X	ist eine der folgenden Zusatzeingaben:
	/R sortiert umgekehrt, alphabetisch von Z nach A, numerisch in absteigender Reihenfolge
	/+n sortiert nach dem Wert in Spalte n

Bemerkungen

Das Programm SORT kann im *Netzwerk* **ohne** Einschränkungen benutzt werden.

Das Programm SORT ist ein *Filter*, der mit anderen Befehlen verkettet werden kann. (Nähere Erläuterungen dazu im Kapitel 8.) Der Filter liest normalerweise Daten vom Standardeingabegerät (Tastatur) und gibt sie auch wieder zum Standardausgabegerät (Bildschirm). Die Ein- oder Ausgabe kann aber auch umgeleitet werden.

Normalerweise wird der Filter SORT mit einem Befehl verbunden. Dies geschieht in der Form:

```
Befehl ¦ SORT
```

Sie haben zusätzlich die Möglichkeit, ein oder zwei Zusatzeingaben, die die Art der Sortierung verändern, mit einzugeben. Die Zusätze werden, durch einen Freiraum getrennt, direkt nach dem Filter eingegeben. Es sind zwei Zusatzeingaben möglich:

/R - umgekehrte Sortierung

Mit diesem Zusatz wird in der umgekehrten Reihenfolge sortiert. Zeichenketten werden jetzt von Z nach A sortiert. Numerische Werte werden jetzt mit der größten Nummer zuerst ausgegeben.

/+n - beginne Sortierung ab Spalte n

Normalerweise wird nach den Zeichen der ersten Spalte sortiert. Durch Eingabe des Zusatzes können Sie andere Spalten als Sortierwerte bestimmen. *n* ist eine beliebige von Ihnen einzugebende Zahl.

Haben Sie zum Beispiel die Datei UNSORT.TXT erstellt und wollen den Inhalt sortiert auf dem Bildschirm darstellen, geben Sie ein:

```
SORT < UNSORT.TXT
```

Soll die Ausgabe in eine andere Datei umgeleitet werden, geben Sie ein:

```
SORT < UNSORT.TXT > B:SORT.TXT
```

Jetzt wird der Inhalt der Datei UNSORT.TXT sortiert in die Datei SORT.TXT auf das Laufwerk B: geschrieben.

Verketten Sie den Filter mit dem Befehl DIR, erhalten Sie ein sortiertes Verzeichnis der Platte. Geben Sie ein:

DIR C:SORT

Das Verzeichnis des Laufwerkes C: wird auf dem Bildschirm sortiert angezeigt. Leiten Sie diese Ausgabe um, z. B. auf den Drucker, geben Sie ein:

DIR C:SORT > PRN

Soll die Ausgabe gleichzeitig auf Bildschirm und Drucker erfolgen, betätigen Sie vor der Befehlseingabe die Tasten **CTRL-PRTSC** (s. a. Kapitel 8). Nach Ausgabe des Verzeichnisses schalten Sie den Drucker mit derselben Tastenkombination wieder aus.

Geben Sie einen Zusatz mit ein, z. B. /+14, wird nicht mehr alphabetisch sortiert, sondern nach dem Wert in der Spalte 14. In dieser Spalte beginnt die *Größenangabe* der Datei in Bytes. Die Dateien werden jetzt nach ihrer Größe sortiert, beginnend mit dem kleinsten Wert. Angenommen, es sind mehr als 20 Dateien im Verzeichnis gespeichert, geben Sie die sortierten Daten in den Filter MORE ein. Die Ausgabe wird dann automatisch unterbrochen, wenn die Seite voll ist (s. a. Seite 11.205). Geben Sie ein:

DIR C:¦SORT /+14¦MORE

Dieses Beispiel läßt sich durch die zusätzliche Eingabe von /R noch leicht verändern, die größten Dateien werden dann zuerst angezeigt.

Fehlermeldungen

SORT: Incorrect DOS Version	SORT: Falsche DOS-Version

ERKLÄRUNG:
Sie benutzen den Filter SORT unter einer DOS-Version die nicht 2.0 oder höher ist.

SORT

SORT:
Insufficient disk space

SORT:
Nicht genug Speicherkapazität
auf der Platte

ERKLÄRUNG:
Bei der Befehlsverkettung mit Filtern werden vorübergehend Dateien auf der Platte angelegt. Dort werden die zu bearbeitenden Daten zwischengespeichert. Auf dem momentanen Arbeitslaufwerk ist nicht mehr genügend freie Speicherkapazität vorhanden für diese Dateien. Nehmen Sie eine andere Diskette, oder löschen Sie nicht mehr benötigte Dateien.

SORT:
Insufficient memory

SORT:
Arbeitsspeicher zu klein

ERKLÄRUNG:
Der freie Arbeitsspeicher ist zu klein. Der Filter SORT kann die Daten nicht korrekt bearbeiten. Starten Sie das System neu, und laden Sie nur die unbedingt notwendigen Programme.

SUBST Extern

Zweck

ersetzt ein Verzeichnis vorübergehend durch einen Laufwerknamen

Syntax

```
[d:] [Weg] SUBST
[d:] [Weg] SUBST d: [d:] Weg
[d:] [Weg] SUBST d: /D
```

dabei bedeutet:

d:	vor dem Programmnamen bestimmt das aufzurufende Laufwerk
Weg	vor dem Programmnamen bestimmt das aufzurufende Verzeichnis
d:	nach dem Programmnamen bestimmt das neue Laufwerk
Weg	nach dem Programmnamen bestimmt das zu ersetzende Verzeichnis in der Form: [\] [Verzeichnis [\Verzeichnis...]]
/D	löscht eine Umbenennung

Bemerkungen

Das Programm SUBST kann im *Netzwerk* **nicht** benutzt werden.

Mit dem Programm JOIN können Sie ein komplettes Laufwerk mit allen Unterverzeichnissen durch einen Verzeichnisnamen ersetzen. Dies kann erforderlich werden, haben Sie sich ein tief gestaffeltes System von Unterverzeichnissen geschaffen. Viele Anwenderprogramme erlauben nur eine bestimmte Zeichenzahl (ca. 30) für den kompletten Dateinamen (Laufwerk, Verzeichnis, Dateiname). Schaffen Sie mit dem Programm JOIN ein neues Verzeichnis, können die Programme wieder problemlos arbeiten.

SUBST

ACHTUNG:
Sie können die folgenden Programme nicht mit einem Laufwerknamen benutzen, der ein Verzeichnis ersetzt:

ASGNPART	FORMAT
ASSIGN	JOIN
BACKUP	LABEL
DISKCOMP	PART
DISKCOPY	RESTORE

Einige ältere Programme können nicht mit dem strukturierten Inhaltsverzeichnis arbeiten, das mit der Version 2.0 eingeführt wurde. Das Programm SUBST ersetzt ein Inhaltsverzeichnis durch eine einfache Laufwerkbezeichnung, mit denen diese Programme arbeiten können. Dies kann ein beliebiger Buchstabe des Alphabetes sein. Den letzten möglichen Buchstaben bestimmen Sie mit dem Befehl LASTDRIVE in der Datei CONFIG.SYS während des Systemstartes. Haben Sie den Befehl nicht benutzt, wird das letzte gültige Laufwerk mit dem Buchstaben E: bezeichnet. Nähere Informationen dazu finden Sie im Kapitel 9.

Geben Sie den Namen des zu ersetzenden Verzeichnisses immer komplett an, indem Sie im Rootdir beginnen. Nach Aufruf des Programmes kann das Verzeichnis aber noch weiterhin mit seinem ursprünglichen Namen benutzt werden.

Damit das Programm SUBST ein Verzeichnis durch ein Laufwerk ersetzen kann, müssen folgende Voraussetzungen eingehalten werden:

- Das Arbeitslaufwerk kann nicht als Ersatz benannt werden.

- Ein Verzeichnis eines umgeleiteten oder eines Netzwerklaufwerkes kann nicht ersetzt werden.

Programmaufruf

Es sind drei verschiedene Befehlseingaben möglich.

1. Rufen Sie das Programm nur mit seinem Namen auf, erhalten Sie eine Übersicht der momentanen Umbenennungen. Geben Sie ein:

   SUBST

   Es könnte z. B. folgende Übersicht erscheinen:

   A: => C:\WORDS
   B: => C:\NUMBERS

2. Ersetzen Sie ein Verzeichnis durch eine Laufwerkbezeichnung. Dabei beachten Sie, daß das momentane Laufwerk nicht als Ersatz für ein Verzeichnis benannt werden darf. Geben Sie ein:

   SUBST A: C:\TEXT

   Jetzt wird das das Verzeichnis \TEXT des Laufwerkes C: durch den Laufwerknamen A: ersetzt. Gleichzeitig kann aber das Verzeichnis \TEXT weiterhin direkt benutzt werden.

3. Löschen Sie eine Umbenennung wieder. Rufen Sie den Befehl mit dem "neuen" Laufwerknamen und dem Zusatz /D auf. Geben Sie ein:

   SUBST A: /D

Fehlermeldungen

Cannot SUBST a Network drive	Ein Verzeichnis des Netzwerkes kann nicht ersetzt werden

ERKLÄRUNG:
Sie haben versucht, mit dem Befehl SUBST ein Verzeichnis eines Netzwerklaufwerkes zu ersetzen. Dies ist nicht möglich.

SUBST

Invalid number of parameters	Falsche Anzahl der Zusatzeingaben

ERKLÄRUNG:
Sie haben den Befehl mit falschen Zusatzeingaben aufgerufen. Im Abschnitt Programmaufruf finden Sie eine Übersicht der drei möglichen Eingabezeilen.

Not enough memory	Arbeitsspeicher zu klein

ERKLÄRUNG:
Das Programm SUBST erweitert das Betriebssystem, damit es eine Tabelle mit den Umbenennungen speichern kann. Der freie Arbeitsspeicher ist für diese Tabelle nicht mehr groß genug. Löschen Sie resident gespeicherte Programme oder reservieren Sie mit dem Befehl BUFFERS weniger Zwischenspeicher.

Invalid path	Falsches Verzeichnis

ERKLÄRUNG:
Sie haben ein nicht vorhandenes Verzeichnis ersetzen wollen. Überprüfen Sie Ihre Eingabe.

SYS Extern

Zweck

kopiert die Systemdateien auf eine Diskette oder Winchester

Syntax

[d:] [Weg] SYS d:

dabei bedeutet:

d:	vor dem Programmnamen bestimmt das aufzurufende Laufwerk
Weg	vor dem Programmnamen bestimmt das aufzurufende Verzeichnis
d:	Ziellaufwerk für die Systemdateien

Bemerkungen

Das Programm SYS kann im Netzwerk nicht benutzt werden.

Mit diesem Befehl werden die beiden *Systemdateien* IO.SYS und MSDOS.SYS kopiert. Dies ist erforderlich, wenn Sie eine neue Version des Betriebssystemes auf vorhandene Disketten kopieren wollen. Sie können mit dem Befehl SYS die Dateien auch auf eine formatierte Diskette kopieren, die noch keine Dateien enthält.

Die Dateien der Ursprungs- und Zieldiskette müssen dabei gleich groß sein, sonst kann der Befehl nicht ausgeführt werden. Wollen Sie Disketten des Systemes 1.25 auf das System 3.0 umstellen, müssen Sie die Diskette neu formatieren. Der Befehl SYS kann diese unterschiedlichen Versionen nicht übertragen.

SYS

Die Dateien werden in der Reihenfolge

IO.SYS
MSDOS.SYS

übertragen. Die Dateien werden anschließend als Systemdateien und verdeckte Dateien markiert. Sie werden dann im Verzeichnis nicht mit aufgelistet und können auch nicht kopiert werden. Die Datei COMMAND.COM wird mit diesem Befehl nicht kopiert.

Der Befehl wird außer einer Eingabe für das Ziellaufwerk ohne weitere Zusätze aufgerufen:

SYS D:

Nach Aufruf des Programmes erscheint die Meldung:

Insert system disk in drive A and any key when ready	Legen Sie die Systemdiskette in das Laufwerk A und betätigen Sie eine Taste

Wurden die Dateien übertragen, erscheint zum Schluß des Programmes die Meldung:

System transferred	System übertragen

Es erscheint wieder das Prompt des Betriebssystemes, und das Programm ist beendet.

Fehlermeldungen

Cannot SYS a Network drive	Auf ein Laufwerk des Netzwerkes können keine Systemdateien kopiert werden

ERKLÄRUNG:
Sie haben versucht, mit dem Befehl SYS ein Laufwerk des Netzwerkes zu bearbeiten. Dies ist nicht möglich.

Incompatible system size	Dateigröße verschieden

ERKLÄRUNG:
Die Systemdateien auf der Ursprungs- und Zieldiskette haben unterschiedliche Größen. Die Dateien können nicht kopiert werden.

Incorrect DOS Version	Falsche DOS-Version

ERKLÄRUNG:
Sie benutzen das Programm SYS unter einer DOS-Version, die nicht 3.0 oder höher ist.

Invalid drive specification	Falsche Laufwerkbezeichnung

ERKLÄRUNG:
Der eingegebene Laufwerkname ist nicht vorhanden oder die Schreibweise war falsch.

Invalid parameter	Falsche Zusatzeingabe

ERKLÄRUNG:
Sie haben außer der Laufwerkangabe weitere Eingaben gemacht. Dies ist nicht erlaubt. Geben Sie den Befehl neu ein.

No room for system on destination disk	Kein Speicherplatz für die Systemdateien auf der Zieldiskette

ERKLÄRUNG:
Auf der Zieldiskette sind bereits Dateien gespeichert. Die Systemdateien können daher nicht mehr an ihren Platz kopiert werden. Formatieren Sie die Diskette für das Format 3.0 neu.

Not ready error reading drive D: Abort, Retry, Ignore?	Lesefehler im Laufwerk D: Abbrechen, Wiederholen, Ignorieren?

ERKLÄRUNG:
Das bezeichnete Laufwerk enthält keine Diskette, oder die Klappe des Laufwerkes ist offen. Beheben Sie den Fehler, und wiederholen Sie den Vorgang mit Eingabe von R. Ist das Laufwerk nicht vorhanden, brechen Sie den Vorgang durch Eingabe von A ab. Geben Sie den Befehl neu ein.

TIME — Intern

Zweck

zeigt und ändert die Uhrzeit

Syntax

```
TIME
TIME hh :mm :ss .cc
```

dabei bedeutet:

hh	Eingabe von 0 bis 23 für die Stunden
mm	Eingabe von 0 bis 59 für die Minuten
ss	Eingabe von 0 bis 59 für die Sekunden
cc	Eingabe von 00 bis 99 für hundertstel Sekunden

Bemerkungen

Die *Uhrzeit* wird vom System bei allen Eingabefunktionen für Dateien benutzt. Damit wird Ihnen immer angezeigt, wann zuletzt in die Datei geschrieben wurde. Sie können den Befehl sowohl direkt eingeben, als auch über eine Batchdatei aufrufen. Rufen Sie die Funktion durch einfache Befehlseingabe auf, erscheint die Meldung:

Current time is hh:mm:ss.cc Enter new time:	Momentane Zeit ist hh:mm:ss.cc Neue Zeit:

Geben Sie dann die Uhrzeit in der Reihenfolge ein:

Stunden	erlaubt sind Zahlen zwischen 0 und 23
Minuten	erlaubt sind Zahlen zwischen 0 und 59
Sekunden	erlaubt sind Zahlen zwischen 0 und 59
hundertstel Sekunden	erlaubt sind Zahlen zwischen 0 und 99

TIME

Die Zahlen dürfen nur durch einen Doppelpunkt (:) getrennt werden. Die Eingabe der Stunden erfolgt im 24-Stunden-Format. Die Zeitanzeige der Dateien erfolgt in der *amerikanischen 12-Stunden-Schreibweise* mit einem nachgestellten „a" für vormittags und einem „p" für nachmittags. Geben Sie hunderstel Sekunden ein, muß die Eingabe durch einen einfachen Punkt getrennt werden. Bei einer falschen Eingabe erscheint die Fehlermeldung:

Invalid time	Falsche Zeit
Enter new time:	Neue Zeit:

Soll die momentane Zeit nicht geändert werden, übernehmen Sie den Wert durch einfaches Betätigen der Taste **RETURN.**

Sie haben die Möglichkeit, die Zeit direkt in der Befehlszeile miteinzugeben. Ein Beispiel:

```
TIME 11:35:00
```

Das System gibt dann keine weitere Meldung aus.

Mit diesem Befehl haben Sie die Möglichkeit, die Stunden nachzuhalten, die Sie für die Bearbeitung einer Datei oder eines Programmes benötigen. Stellen Sie zu Beginn Ihrer Arbeit die interne Uhr auf null durch Eingabe von

```
TIME 0
```

Rufen Sie am Ende der Arbeit die Zeit durch Eingabe des Befehls ab, sehen Sie die Zeit, die seit dem Beginn verflossen ist.

TREE Extern

Zweck

zeigt das strukturierte Verzeichnis einer Diskette oder Winchester mit Dateien.

Syntax

[d:] [Weg] TREE ?
[d:] [Weg] TREE [d:] [/F]

dabei bedeutet:

d:	vor dem Programmnamen bestimmt das aufzurufende Laufwerk
Weg	vor dem Programmnamen bestimmt das aufzurufende Verzeichnis
?	es erscheint eine Übersicht mit Erläuterungen
d:	Name des Laufwerkes mit der Diskette
/F	es werden alle Dateien im Verzeichnis gezeigt

Bemerkungen

Das Programm TREE kann im Netzwerk ohne Einschränkungen benutzt werden.

Das Dienstprogramm TREE zeigt schnell und übersichtlich sämtliche Verzeichnisse einer Platte an. Damit wird die hierarchische Struktur des Inhaltsverzeichnisses deutlich. Für jedes Unterverzeichnis wird der komplette Name angezeigt. Da für das erste Verzeichnis der Platte (Rootdir) kein Name vorhanden ist, wird hierfür ein umgekehrter Schrägstrich (\) gezeigt. Geben Sie den Zusatz /F mit ein, werden zusätzlich sämtliche Dateien im jeweiligen Verzeichnis mit aufgelistet. Das Rootdir kann mit diesem Befehl nicht dargestellt werden. Benutzen Sie dafür den Befehl DIR. Sind Sie nicht im Rootdir, geben Sie den Befehl mit dem Schrägstrich (\) ein (s. a. Seite 11.40).

Sind auf dem bezeichneten Laufwerk keine Unterverzeichnisse vorhanden, wird das Programm sofort mit der Meldung beendet:

No sub-directories exist	Es sind keine weiteren Verzeichnisse vorhanden

Die Ausgabe des Programmes kann umgeleitet werden, z. B. in eine Datei oder auf den Drucker. Leiten Sie die Ausgabe erst in den Filter MORE, wird das Verzeichnis seitenweise auf dem Bildschirm angezeigt. Es besteht die Möglichkeit, mit Hilfe der Tastenkombination **SHIFT-PRTSC** dann gezielt einzelne Seiten auszudrucken.

Programmaufruf

Sie können jederzeit eine kurze Erklärung des Programmes TREE aufrufen. Geben Sie den Befehl ein:

```
TREE ?
```

Es erscheint dann folgende Übersicht:

```
                 TREE Version 3.01
Copyright (C) 1985, Zenith Data Systems Corporation

TREE shows the subdirektory structure of the default or specified disk. If the optional F
(Files) switch is used. TREE also lists the files contained in each subdirectory.

Syntax: TREE ?
        TREE [d:] [/F]
```

Rufen Sie das Programm für das momentane Laufwerk auf, geben Sie ein:

```
TREE
```

TREE

Die Übersicht der Verzeichnisse wird in folgender Form wiedergegeben:

```
DIRECTORY PATH LISTING FOR VOLUME Bezeichnung

PATH: Name

Sub-directories: Name
                 .
                 .
                 Name

Files:           Dateiname
                 .
                 .
                 Dateiname
```

dabei bedeutet:

Bezeichnung	Name der Diskette (falls beim Formatieren eingegeben)
PATH	Liste der Unterverzeichnisse vom Rootdir an
Sub-directory	Name des Unterverzeichnisses
Files	Name der Datei (wird nur bei Eingabe von /F angezeigt)

Ist für die Platte kein Name gespeichert worden, wird dafür in der ersten Zeile

```
DIRECTORY PATH LISTING FOR VOLUME ???????????
```

angezeigt. Enthält ein Unterverzeichnis keine weiteren Verzeichnisse, wird dies durch folgende Zeile angezeigt:

```
Sub-directories: none
```

Haben Sie das Programm mit dem Zusatz /F aufgerufen in der Form

TREE /F

werden alle Dateien des Verzeichnisses mit aufgelistet. Enthält ein Verzeichnis keine Dateien, wird dies durch folgende Zeile angezeigt:

Files: none

Soll das strukturierte Verzeichnis nicht für das aktuelle Laufwerk angezeigt werden, rufen Sie das Programm auf mit dem Befehl:

TREE d:

Dabei ist *d:* eine gültige Laufwerkbezeichnung. Ebenso ist es möglich, auch hier den Zusatz /F mit einzugeben. Der Zusatz kann vor oder nach der Laufwerkbezeichnung, mit oder ohne Leerzeichen eingegeben werden. Alle Schreibweisen sind richtig und werden vom Programm akzeptiert.

Beispiel

Die Beispieldiskette enthält folgende Verzeichnisse und Dateien:

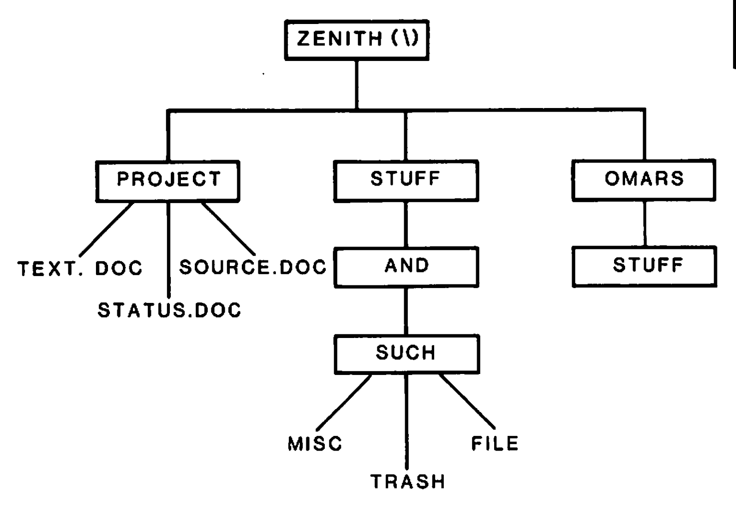


Bild 11.13
Strukturiertes Inhaltsverzeichnis

TREE

Das Programm TREE zeigt alle Verzeichnisse und Dateien nach Eingabe des Befehls auf dem Bildschirm in folgender Weise an. Geben Sie ein:

TREE /F

```
DIRECTORY PATH LISTING FOR VOLUME ZENITH

Path: \PROJEKT

Sub-directories: None

Files:           TEXT   .DOC
                 SOURCE .DOC
                 STATUS .DOC

Path: \STUFF

Sub-directories: AND

Files:           None

Path: \STUFF\AND

Sub-directories: SUCH

Files:           None

Path: \STUFF\AND\SUCH

Sub-directories: None

Files:           MISC
                 TRASH
                 FILE

Path: \OMARS

Sub-directories: STUFF

Files:           None

Path: \OMARS\STUFF

Sub-directories: None

Files:           None
```

Soll die Ausgabe auf den Drucker umgeleitet werden, geben Sie folgenden Befehl ein:

TREE /F > PRN

Fehlermeldungen

Incorrect DOS Version — Falsche DOS-Version

ERKLÄRUNG:
Sie benutzen das Programm TREE unter einer DOS-Version, die nicht 3.0 oder höher ist.

Invalid parameter — Falsche Zusatzeingabe

ERKLÄRUNG:
Es ist nur der Zusatz /F oder ? erlaubt. Alle anderen Eingaben rufen diese Fehlermeldung hervor. Geben Sie den Befehl neu ein.

TYPE **Intern**

Zweck

zeigt den Inhalt einer ASCII-Datei auf dem Bildschirm an

Syntax

TYPE Dateibez
TYPE [D:] [Weg] Dateiname

dabei bedeutet:

Dateibez genaue Bezeichnung der anzuzeigenden Datei

D: Name des Laufwerks, auf dem sich die Datei befindet

Weg mehrere Verzeichnisse in Folge in der Form:

[\] [Verzeichnis [\Verzeichnis...]]

Dateiname Name der zu zeigenden Datei

Bemerkungen

Dieser Befehl schreibt den kompletten *Dateiinhalt* auf den Bildschirm. Es wird jedes Zeichen dargestellt. Der Befehl sollte nur für Textdateien eingegeben werden. Verwenden Sie den Befehl für binäre Dateien (.COM oder .EXE), werden auch Kontrollzeichen dargestellt. Der Inhalt von binären Dateien kann vom Programm DEBUG besser angezeigt werden. Der Inhalt der Datei wird mit dem Befehl TYPE nicht verändert. Dafür ist ein Textverarbeitungsprogramm erforderlich oder das Dienstprogramm EDLIN.

Verwenden Sie den Befehl für größere Dateien, kann der gesamte Text nicht auf dem Bildschirm dargestellt werden. Der erste Teil des Textes wird dann automatisch zeilenweise aus dem Bild gerollt. Mit der Tastenkombination **CTRL-S** oder **CTRL-NUM LCK** kann die Ausgabe vorübergehend angehalten werden. Eleganter ist es, den Befehl mit dem Filter MORE zu verketten. Der Filter MORE stoppt die Ausgabe automatisch nach 23 Zeilen.

Ebenso ist es möglich, die Ausgabe des Befehls TYPE umzuleiten, z. B. auf einen Drucker. Nähere Informationen dazu im Kapitel 8. Wollen Sie mehrere Dateien nacheinander anzeigen lassen, können Sie den Befehl TYPE auch mit dem Programm APPLY verbinden.

Soll eine Datei des momentanen Verzeichnisses auf dem Bildschirm angezeigt werden, geben Sie den kompletten Dateinamen nach dem Befehl ein. Ein Beispiel:

```
TYPE AUTOEXEC.BAT
```

Jetzt wird der Inhalt der Datei AUTOEXEC.BAT auf dem Bildschirm angezeigt. Ist die Datei nicht im momentanen Verzeichnis oder nicht auf dem momentanen Laufwerk gespeichert, müssen Sie den Laufwerknamen und/oder das Verzeichnis vor dem Dateinamen eingeben. Ein Beispiel:

```
TYPE F: \TEXT\SPEICHER\BRIEF.PRV
```

Die Datei BRIEF.PRV ist auf dem Laufwerk F: im Verzeichnis TEXT SPEICHER gespeichert. Auch diese Ausgabe können Sie umleiten, indem Sie nicht den Bildschirm als Ausgabegerät benennen, sondern z. B. den Drucker (PRN). Geben Sie ein:

```
TYPE F: \TEXT\SPEICHER\BRIEF.PRV > PRN
```

TYPE

Fehlermeldungen

File not found — Datei nicht vorhanden

ERKLÄRUNG:
Der eingegebene Dateiname ist im Verzeichnis nicht vorhanden. Überprüfen Sie Ihre Eingabe.

Invalid number of parameters — Falsche Anzahl der Zusatzeingaben

ERKLÄRUNG:
Sie haben den Befehl ohne Dateiname eingegeben. Überprüfen Sie Ihre Eingabe.

Not ready error reading drive D:
Abort, Retry, Ignore?

Lesefehler im Laufwerk D:
Abbrechen, Wiederholen, Ignorieren?

ERKLÄRUNG:
Das bezeichnete Laufwerk enthält keine Diskette, oder die Klappe des Laufwerkes ist offen. Beheben Sie den Fehler, und wiederholen Sie den Vorgang mit Eingabe von R. Ist das Laufwerk nicht vorhanden, brechen Sie den Vorgang durch Eingabe von A ab. Geben Sie den Befehl neu ein.

VER **Intern**

Zweck

zeigt die Versionsnummer der Dateien IO.SYS und MSDOS.SYS an.

Syntax

```
VER
```

Bemerkungen

Nach Eingabe des Befehls wird auf dem Bildschirm die *Versionsnummer* der beiden Dateien IO.SYS und MSDOS.SYS angezeigt. Die Nummer setzt sich zusammen aus einer einzelnen *Ziffer (Hauptnummer)* und zwei *Folgeziffern* (Unternummer), die durch einen Punkt getrennt werden. Nach Eingabe von:

```
VER
```

erscheint beispielsweise:

```
IO.SYS Version 3.03
MS-DOS Version 3.10
```

VERIFY Intern

Zweck

überprüft kopierte Daten

Syntax

VERIFY [ON]
VERIFY [OFF]

Bemerkungen

Dieser Befehl ruft dieselbe Funktion auf, wie der Zusatz /V des Befehls COPY.

Das Betriebsystem *vergleicht* die gerade geschriebenen Daten mit den Ursprungsdaten. Wird hierbei ein Unterschied festgestellt, wird eine Fehlermeldung ausgegeben. Mit diesem Befehl rufen Sie die Funktion unabhängig von einem momentanen Kopiervorgang auf. Sie bleibt solange eingeschaltet, bis sie mit dem Befehl wieder ausgeschaltet wird.

Sie geben den Befehl ein mit dem Zusatz für die jeweilige Funktion:

VERIFY ON — VERIFY wird eingeschaltet

oder

VERIFY OFF — VERIFY wird ausgeschaltet

Geben Sie den Befehl ohne Zusatz ein

VERIFY

wird der momentane Status angezeigt. Dies ist entweder

VERIFY is on — VERIFY ist eingeschaltet

oder

VERIFY is off — VERIFY ist ausgeschaltet

VOL Intern

Zweck

zeigt den Namen des Laufwerkes an (falls vorhanden)

Syntax

VOL D:

Bemerkungen

Mit diesem Befehl kann jederzeit der *Name* der Diskette oder der Winchester angezeigt werden. Geben Sie keine Laufwerkbezeichnung ein, wird der Name des Arbeitslaufwerkes angezeigt. Der *Laufwerkname* kann von Ihnen eingegeben werden, wenn Sie die Platte formatieren oder mit dem Programm LABEL arbeiten. Er darf aus maximal elf Zeichen bestehen. Es gelten dieselben Regeln wie für Dateinamen (s. a. Kapitel 1).

Geben Sie den Befehl ein, erscheint folgender Text:

Volume in drive D is ....

Der Name wird immer in Großbuchstaben angezeigt.

Haben Sie beim Formatieren der Platte keinen Namen eingegeben, erscheint folgender Text:

Volume in drive D has no Label   Laufwerk D hat keinen Namen

Teil IV

Weitere Dienstprogramme

Weitere Dienstprogramme

Dieser Teil erläutert ausführlich vier weitere Dienstprogramme, die zur Programmentwicklung und Herstellung benutzt werden. Benutzer des Z-100 PC, die nicht selber Programme schreiben wollen, brauchen diese Kapitel nicht zu lesen.

Die Erklärungen erfolgen teilweise mit Fachausdrücken, die ein spezielles Wissen über den Prozessor voraussetzen. Ebenso sollten allgemeine Programmierkenntnisse vorhanden sein. Außerdem werden die in den ersten Kapiteln erklärten Begriffe als bekannt vorausgesetzt.

Die Reihenfolge der Befehlserklärungen erfolgt so, wie diese im Programm wahrscheinlich auch von Ihnen benutzt werden. Trotzdem kann es vorkommen, daß Sie die Programme in einer anderen Reihenfolge benutzen müssen.

Dieser Teil des Handbuches besteht aus den folgenden vier Kapiteln:

Kapitel 12: EDLIN erklärt die Befehlseingaben und den Ablauf des Programmes, daß Sie zur Herstellung von Quelldateien benutzen können.

Kapitel 13: LIB erklärt die Befehlseingaben und den Ablauf des Programmes, das Bibliotheken erstellt und verwaltet.

Kapitel 14: LINK erklärt die Befehlseingabe und den Ablauf des Programmes, das einzelne Module zu einem ablauffähigen Programm verbindet.

Kapitel 15: EXE2BIN erklärt die Befehlseingabe und den Ablauf des Programmes, das eine EXE-Datei in eine COM-Datei umformt.

Kapitel 16: DEBUG erklärt die Befehlseingaben und den Ablauf des Programmes, mit dem Sie fertige Programme testen und ändern können.

Für die Befehlseingabe dieser externen Befehle gelten die in Kapitel 5 erläuterten Vorschriften.

EDLIN

Übersicht

EDLIN ist das MS-DOS Dienstprogramm zur Herstellung und Änderung von *Textdateien* und *Quellprogrammen*. Es ist auf der ersten Systemdiskette enthalten. Das Programm EDLIN wird auch Zeileneditor genannt, weil der Text in Zeilen von maximal 253 Zeichen unterteilt ist. Zur Änderung einzelner Zeilen und bestimmter Zeilengruppen können die Funktions- und Kontrolltasten (s. a. Kapitel 5) benutzt werden.

Die Vergabe von Zeilennummern erfolgt automatisch von 1 bis 65533. Fügen Sie neue Zeilen ein, werden die Nummern jeweils um eins erhöht. Ebenso werden die Nummern um eins vermindert, wenn Sie Zeilen löschen. Die Zeilennummern werden nicht als Teil der Datei gespeichert.

Mit dem Programm EDLIN können folgende Funktionen ausgeführt werden:

- Erstellen und Speichern von neuen Text- und Programmdateien
- Überarbeiten bestehender Dateien, alte und neue Fassung speichern
- Anzeigen, Ändern, Einfügen und Löschen von Zeilen
- Suchen, Ersetzen und Löschen von Zeichenketten in einer oder mehreren Zeilen
- Suchen, Ersetzen und Löschen von Steuerzeichen in einer oder mehreren Zeilen
- Abarbeitung gespeicherter Befehle (Befehlsmakro)

Programmaufruf

[d:] [Weg] EDLIN Dateibez [/B] < Eingabe

dabei bedeutet:

d:	vor dem Programmnamen bestimmt das aufzurufende Laufwerk
Weg	vor dem Programmnamen bestimmt das aufzurufende Verzeichnis
Dateibez	Name der zu bearbeitenden Datei mit Laufwerksname und/oder Verzeichnisname
/B	lädt die Datei binär (ignoriert CTRL-Z)
< **Eingabe**	Textdatei, aus der alle Eingaben des Programmes gelesen werden sollen

Bemerkungen

Sie rufen das Programm zusammen mit dem Namen der zu bearbeitenden Datei auf. Existiert die Datei nicht, wird eine neue angelegt. Geben Sie ein:

EDLIN Dateibez

Zum Schluß des Programmes ist es nicht mehr möglich, die hier eingegebene Dateibezeichnung zu ändern. Die bearbeitete Datei wird immer auf das Laufwerk und in das Verzeichnis geschrieben, das beim Aufruf benannt wurde.

Da das Programm EDLIN Textdateien bearbeitet, sucht es immer nach dem Zeichen für *Dateiende* (EOF). Dies ist ein CTRL-Z (Wert 1Ah). Benutzen Sie dieses Zeichen im Text, würde der Rest der Datei nicht mehr gelesen. Rufen Sie das Programm EDLIN mit dem Zusatz /B auf, wird die Datei jedoch mit der Zahl der Bytes, die im Inhaltsverzeichnis gespeichert ist, geladen.

Programmaufruf

Das Programm EDLIN liest die *Standardeingabe* des Betriebssystemes. Damit kann sehr einfach die Funktion der Datenumleitung benutzt werden. Speichern Sie alle auszuführenden Befehle des Programmes in einer Textdatei. Teilen Sie dann dem Programm beim Aufruf in der Befehlszeile durch das Zeichen „<" mit, daß die folgende Datei alle Befehle enthält. Sie können damit Befehlsmakros arbeiten lassen. Vergessen Sie jedoch nie, als letzten Befehl Q(uit) einzugeben. Bei Datenumleitung ist keine Eingabe über die Tastatur mehr möglich.

Nach dem Aufruf sucht das Programm nach dem zur Bearbeitung eingegebenen Dateinamen. Findet es diesen nicht, meldet es sich mit seinem Anforderungszeichen. Das ist hier ein Stern (*). Es erscheint dann die Meldung:

New File	Neue Datei
*	*

Findet EDLIN den zur Bearbeitung eingegebenen Dateinamen, wird diese Datei in den Arbeitsspeicher geladen. Wurde die gesamte Datei geladen, erscheint die Meldung:

End of input file	Gesamte Datei geladen
*	*

Ist der Arbeitsspeicher für die gesamte Datei zu klein, wird er nur zu maximal 2/3 mit der Datei gefüllt. Mit den Befehlen A(ppend) und W(rite) kann der Rest der Datei bei Bedarf nachgeladen werden. In diesem Falle erscheint kein Meldetext, sondern nur der Stern:

*	*

Dateien mit dem Zusatz .BAK oder .$$$ dürfen nicht aufgerufen werden. Wollen Sie diese Dateien bearbeiten, müssen Sie sie vorher umbenennen.

Programmaufruf

Nach dem Aufruf des Dienstprogrammes EDLIN wird eine Arbeitsdatei mit dem Zusatz .$$$ angelegt. In diese Datei schreibt das Programm alle laufenden Daten. Zum Programmschluß wird die Arbeitsdatei mit dem richtigen Namen umbenannt und die alte Originaldatei erhält den Zusatz .BAK. Die alte Datei mit dem Zusatz .BAK wird gelöscht. Sollte der Platz auf der Diskette für alle Dateien nicht ausreichen, wird zwischendurch die alte Sicherungsdatei gelöscht.

Sie können das Programm auf zwei verschiedene Arten beenden. Mit dem Befehl Q(uit) verlassen Sie das Programm, ohne die vorgenommenen Änderungen zu speichern. Mit dem Befehl E(nd) werden die neuen Daten in eine neue Datei geschrieben.

Befehle

Zum Bearbeiten der Datei können Sie im Programm EDLIN zwei verschiedene Befehlsarten benutzen:

- Befehle, die die Eingabe in einer Zeile beeinflussen
- Befehle, die die Eingabe einer ganzen Zeile beeinflussen

Die Tabelle 12.1 zeigt eine Übersicht der Befehle, die eine ganze Zeile beeinflussen. Die Befehle bestehen aus einem Buchstaben mit optional einzugebenden Zusätzen (s. Tabelle 12.2).

Geben Sie einen Befehl falsch ein, wird dieser nicht ausgeführt. Es erscheint die Fehlermeldung:

Entry error	Eingabefehler

Überprüfen Sie die Eingabe und korrigieren Sie die Zeile mit Hilfe der Funktionstasten.

Geben Sie mehrere Befehle in einer Zeile ein, werden die Befehle durch ein Semikolon (;) getrennt. Sie werden nacheinander abgearbeitet.

Tabelle 12.1 EDLIN-Befehle

BEFEHL	SCHREIBWEISE
Append Lines	[Anzahl] A
Copy Lines	[ZeileA], [ZeileE], ZeileN, [Anzahl] C
Delete Lines	[ZeileA] [,ZeileE] D
Edit Line	Zeile
End Edit	E
Insert Line	Zeile I
List Lines	[ZeileA] [,ZeileE] L
Move Lines	[ZeileA], [ZeileE], ZeileN M
Page Text	[ZeileA], [ZeileE] P
Quit Edit	Q
Replace Text	[ZeileA] [,ZeileE] [?] R [Zeichenkette1] F6 [Zeichenkette2]
Search Line	[ZeileA] [,ZeileE] [?] S [Zeichenkette]
Transfer Lines	[ZeileN] T Dateiname
Write Lines	[Anzahl] W

Befehlszusätze

Zu den in der Tabelle 12.1 aufgelisteten Befehlen können Sie die in der Tabelle 12.2 beschriebenen Zusätze eingeben. Durch diese Zusätze bestimmen Sie die Funktion des Befehls näher.

Befehle

Tabelle 12.2 EDLIN-Variable

VARIABLE	ERKLÄRUNG
Anzahl	numerischer Wert
ZeileX	Eingabe einer Zeilennummer: ZeileA = Anfangszeile ZeileE = Endzeile ZeileN = Zeilennummer Werden mehrere Zeilennummern nacheinander eingegeben, müssen diese durch ein Komma oder eine Leertaste getrennt werden. Zeilennummern können folgendermaßen eingegeben werden: 1. Durch eine *Zahl* von 1 bis 65533. Ist die Nummer größer als die letzte gespeicherte Zeile, wird die neue Zeile automatisch an das Dateiende gesetzt. 2. Durch Eingabe des *Punktes* (.). Damit wird die momentane Arbeitszeile aufgerufen. Diese wird immer durch den Stern (*) gekennzeichnet. 3. Durch Eingabe des *Numeruszeichens* (#). Damit wird die Zeilennummer nach der letzten Zeile der Datei aufgerufen. 4. Durch **RETURN**. Damit wird der vom Programm EDLIN vorgegebene Wert übernommen (meistens die Arbeitszeile)

Tabelle 12.2 EDLIN-Variable

VARIABLE	ERKLÄRUNG
?	Eingabe bei den *Zeilenbefehlen* R(eplace) und S(earch). Die Bearbeitung der gefundenen Zeichenkette wird erst vorgenommen, wenn ein **Y** für ja oder **RETURN** eingegeben wird. Die Zeichenkette wird mit einem **N**(ein) nicht verändert.
<Zeichen-kette>	Eingabe eines Textes, der gesucht oder ersetzt werden soll. Die Eingabe des Textes wird durch ein **CTRL-Z** oder **RETURN** abgeschlossen. Leerfelder zwischen den Zeichenketten oder Zeilenbefehlen dürfen nicht eingegeben werden, es sei denn, sie sind Teil der Zeichenkette.

Append

Zweck

lädt weitere Zeilen der zu bearbeitenden Datei in den Arbeitsspeicher

Syntax

[Anzahl] A

dabei bedeutet:

Anzahl Zahl der zu ladenden Zeilen

Bemerkungen

Ist die zu bearbeitende Datei zu groß, um auf einmal in den Speicher kopiert zu werden, wird dies beim Programmaufruf dadurch angezeigt, daß sofort das Prompt erscheint.

Sie können dann den ersten schon geladenen Teil der Datei bearbeiten. Ist die Bearbeitung abgeschlossen, muß dieser Teil der Datei auf die Platte zurückgeschrieben werden. Dies geschieht mit dem Befehl W(rite). Jetzt können weitere Zeilen der Datei in den wieder freigewordenen Speicher kopiert werden. Geben Sie den Befehl **A** ohne Zusatz ein, wird vom Programm EDLIN soviel in den Speicher kopiert, bis die Datei komplett im Speicher ist, oder dieser zu maximal 2/3 gefüllt ist. Geben Sie eine Zeilenzahl vor dem Befehl ein, wird die Anzahl der Zeilen nachgeladen. Ein Beispiel:

125 A

Es werden jetzt maximal 125 Zeilen der Datei von der Platte nachgeladen. Ist damit die Datei komplett nachgeladen, erscheint die Meldung:

End of input file	Gesamte Datei geladen
*	*

Ist die Datei noch nicht komplett geladen, erscheint das Prompt wiederum ohne Meldetext.

Copy

Zweck

kopiert bestimmte Zeilen an eine angegebene Stelle der Datei

Syntax

[ZeileA], [ZeileE], ZeileN, [Anzahl] C

dabei bedeutet:

ZeileA Anfangszeile, das ist die erste Zeile, die kopiert wird.

ZeileE Endzeile, das ist die letzte Zeile, die kopiert wird.

ZeileN Zeilennummer, das ist die Zeile, vor der der kopierte Text eingefügt wird.

Anzahl gibt an, wie oft die bezeichneten Zeilen kopiert werden sollen.

Bemerkungen

Geben Sie die Anfangs- oder Endzeile nicht mit ein, wird der Wert dafür automatisch vom Programm EDLIN eingesetzt. Dies ist dann die durch den Stern gekennzeichnete Arbeitszeile. Wird die Anzahl der Kopiervorgänge nicht angegeben, werden die Zeilen einmal kopiert. Nach dem Kopiervorgang wird die Datei automatisch neu numeriert.

Überschneiden sich die zu kopierenden Zeilen, führt das Programm den Befehl nicht aus, und es erscheint die Fehlermeldung:

Entry error Eingabefehler

Copy

Für das Beispiel wird folgender Text benutzt:

1: Dieses ist eine Testdatei,
2: die zeigt, wie mit dem Programm gearbeitet wird,
3: die zeigt, was passiert,
4: wenn herumgespielt wird.
5: Es gibt zwei sehr brauchbare Befehle
6: zum Verändern der Datei:
7:*COPY und INSERT.

Kopieren Sie den gesamten Text mit der Eingabe von:

1,7,8,1 C

Damit kopieren Sie den Text von Zeile 1 bis Zeile 7 einmal. Der Text wird vor der Zeile 8 eingefügt. Die neue Datei sieht dann folgendermaßen aus:

```
 1: Dieses ist eine Testdatei,
 2: die zeigt, wie mit dem Programm gearbeitet wird,
 3: die zeigt, was passiert,
 4: wenn herumgespielt wird.
 5: Es gibt zwei sehr brauchbare Befehle
 6: zum Verändern der Datei:
 7: COPY und INSERT.
 8: Dieses ist eine Testdatei,
 9: die zeigt, wie mit dem Programm gearbeitet wird,
10: die zeigt, was passiert,
11: wenn herumgespielt wird.
12: Es gibt zwei sehr brauchbare Befehle
13: zum Verändern der Datei:
14:*COPY und INSERT.
```

Im folgenden Beispiel werden die Zeilen 1 bis 2 kopiert und vor der Zeile 5 wieder eingefügt:

1,2,5,1 C

Die Datei wird wieder automatisch umnumeriert und sieht so aus:

```
1: Dieses ist eine Testdatei,
2: die zeigt, wie mit dem Programm gearbeitet wird,
3: die zeigt, was passiert,
4: wenn herumgespielt wird.
5: Dieses ist eine Testdatei,
6: die zeigt, wie mit dem Programm gearbeitet wird,
7: Es gibt zwei sehr brauchbare Befehle
8: zum Verändern der Datei:
9: COPY und INSERT.
```

Delete

Zweck

es werden eine oder mehrere Zeilen der Datei gelöscht

Syntax

[ZeileA] [,ZeileE] D

dabei bedeutet:

ZeileA	Anfangszeile, das ist die erste Zeile, die gelöscht wird.
ZeileE	Endzeile, das ist die letzte Zeile, die gelöscht wird.

Bemerkungen

Mit diesem Befehl löschen Sie eine oder mehrere bezeichnete Zeilen der Datei. Danach wird die Datei sofort automatisch neu numeriert. Die neue Arbeitszeile ist die erste Zeile nach dem gelöschten Block. Geben Sie für ZeileA und für ZeileE Werte ein, werden diese beiden Zeilen und alle dazwischen befindlichen gelöscht. Soll nur eine einzelne Zeile gelöscht werden, geben Sie nur diese Nummer ein. Soll von der Arbeitszeile bis zur bezeichneten Zeile gelöscht werden, kann die erste Zeilenangabe entfallen. Die Befehlseingabe beginnt dann mit einem Komma.

Einzelne Zeile löschen

Die Beispieldatei sieht folgendermaßen aus:

1: Dieses ist eine Testdatei,
2: die zeigt, wie mit dem Programm gearbeitet wird,
3: die zeigt, was passiert,
4: wenn herumgespielt wird.
5: Es gibt zwei sehr brauchbare Befehle
6: zum Verändern der Datei:
7:*Delete and Insert,
8: (der Befehl D und I),
9: (Der Befehl CTRL-BREAK beendet den Modus Einfügen),
10: ändern Zeilennummern.
11: Der Befehl Insert kann irgendwo in der Datei
12: Zeilen einfügen; es gibt keine
13. Probleme bei der Numerierung,
14. da diese dynamisch bis 65533 geführt wird.

Es soll aus dieser Datei die Zeile 10 gelöscht werden. Geben Sie ein:

10 D

Die Datei sieht jetzt so aus:

1: Dieses ist eine Testdatei,
2: die zeigt, wie mit dem Programm gearbeitet wird,
3: die zeigt, was passiert,
4: wenn herumgespielt wird.
5: Es gibt zwei sehr brauchbare Befehle
6: zum Verändern der Datei:
7: Delete and Insert,
8: (der Befehl D und I),
9: (Der Befehl CTRL-BREAK beendet den Modus Einfügen),
10:*Der Befehl Insert kann irgendwo in der Datei
11: Zeilen einfügen; es gibt keine
12: Probleme bei der Numerierung,
13: da diese dynamisch bis 65533 geführt wird.

Delete

Mehrere Zeilen löschen

Aus der geänderten Beispieldatei (s. o.) sollen jetzt die Zeilen 5 bis 10 gelöscht werden. Geben Sie dafür ein:

```
5,10 D
```

Die neue Datei sieht dann so aus:

```
1:  ieses ist eine Testdatei,
2:  ie zeigt, wie mit dem Programm gearbeitet wird,
3:  ie zeigt, was passiert,
4:  wenn herumgespielt wird.
5:*Zeilen einfügen; es gibt keine
6:  Probleme bei der Numerierung,
7:  da diese dynamisch bis 65533 geführt wird.
```

Sollen mehrere nicht zusammenhängende Zeilen gelöscht werden, so muß jede Zeile einzeln gelöscht werden.

Edit

Zweck

kopiert eine zu bearbeitende Zeile in den Tastatureingabespeicher

Syntax

ZeileN

Bemerkungen

Durch Eingabe einer vorhandenen Zeilennummer kopieren Sie diese Zeile vom Arbeitsspeicher in den Tastatureingabespeicher. Diese Zeile wird dann auf dem Bildschirm angezeigt und kann mit Hilfe der Funktionstasten verändert werden. Daß dies jetzt die Arbeitszeile ist, wird durch den Stern nach der Zeilennummer kenntlich gemacht.

Mit Betätigen der Taste **RETURN** rufen Sie die Zeile nach der letzten Arbeitsszeile auf.

Betätigen Sie sofort nach Aufruf der Zeile die Taste **RETURN** ohne weitere Eingaben, wird die Zeile unverändert übernommen. Betätigen Sie die Taste in der Mitte einer Zeile, werden sämtliche Zeichen nach dem Cursor gelöscht. Ein Beispiel:

1: **Dieses ist eine Testdatei,**
2: **die zeigt, wie mit dem Programm gearbeitet wird,**
3: **die zeigt, was passiert,**
4: **wenn herumgespielt wird.**

Rufen Sie die Zeile 1 auf durch Eingabe von:

1

so erscheint auf dem Bildschirm folgendes:

```
1: *Dieses ist eine Testdatei,
1: *_
```

End

Zweck

beendet das Programm, sichert die alte und neue Datei

Syntax

E

Bemerkungen

Zu diesem Befehl können Sie keine weiteren Eingaben machen. Die gerade bearbeitete Datei wird mit dem aufgerufenen Namen auf das beim Aufruf bezeichnete Laufwerk geschrieben. Die alte Datei mit gleichen Namen (falls keine neue Datei bearbeitet wurde) erhält jetzt den Zusatz .BAK. Die alte Datei mit dem Zusatz .BAK (falls vorhanden) wird gelöscht.

Achten Sie darauf, daß das Laufwerk, auf das die neue Datei geschrieben werden soll, noch über genügend freie Speicherkapazität verfügt. Wird die gerade bearbeitete Datei gesichert und ist nicht mehr ausreichend Speicherplatz vorhanden, kann nur ein Teil auf Platte geschrieben werden. Der Rest ist verloren.

Insert

Zweck

fügt Zeilen in die Datei ein

Syntax

ZeileN I

dabei bedeutet:

Zeile N	Zeilennummer, das ist die Zeile, vor der die eingegebene Zeile eingefügt wird.

Bemerkungen

Mit diesem Befehl werden eine oder mehrere Zeilen in die Datei eingefügt. Erstellen Sie eine neue Datei, müssen Sie mit diesem Befehl beginnen.

Die neu einzufügenden Zeilen werden immer vor der letzten Arbeitszeile in die Datei eingefügt. Bei einer neuen Datei genügt es, den Befehl ohne Zusätze einzugeben.

Der Modus *Einfügen* (Insert) bleibt solange erhalten, bis Sie keinen weiteren Text mehr eingeben wollen und zu Beginn einer Zeile die Bearbeitung mit **CTRL-C** oder **CTRL-Z** und **RETURN** beenden. Danach ist die erste Zeile nach dem eingegebenen Text die momentane Arbeitszeile.

Text vor einer bezeichneten Zeile einfügen

Die Beispiele werden an folgender Datei erklärt:

1: Dieses ist eine Testdatei,
2: die zeigt, wie mit dem Programm gearbeitet wird,
3: die zeigt, was passiert,
4: wenn herumgespielt wird.

Werden in diesem Text vor der Zeile 3 weitere Zeilen eingefügt, geben Sie dafür ein:

```
3 I
```

es erscheint jetzt folgendes:

```
3:*_
```

Geben Sie jetzt den Text

```
wie man Zeilen einfügen kann und
```

ein. Beenden Sie die Eingabe mit **RETURN**. Danach zeigt der Bildschirm folgendes:

```
3: wie man Zeilen einfügen kann und
4:*_
```

Beenden Sie den Modus *Einfügen* jetzt durch die Tastenkombination **CTRL-C**. Den neuen Text können Sie dann mit dem Befehl L(ist) auflisten. Er sieht so aus:

```
1: Dieses ist eine Testdatei,
2: die zeigt, wie mit dem Programm gearbeitet wird,
3: wie man Zeilen einfügen kann und
4:*die zeigt, was passiert,
5: wenn herumgespielt wird.
```

Text vor der Arbeitszeile einfügen

Im letzten Beispiel ist die Zeile 4 zur Arbeitszeile geworden (gekennzeichnet durch den Stern). Soll direkt vor dieser Zeile eine weitere eingefügt werden, geben Sie ein **I** ein. Der geforderte Wert der Zeilennummer wird vom Programm durch die Arbeitszeile ersetzt und es erscheint dann:

```
4:*_
```

Der Modus *Einfügen* kann nach der Eingabe beliebiger Zeilen durch die Tastenkombination **CTRL-C** beendet werden. Mit dieser Kombination brechen Sie den Vorgang ab. Damit wird die letzte Zeile nicht mehr in den Text eingefügt.

Die Tastenkombination **F6-RETURN** sollte hier nur dann verwendet werden, wenn auch gleichzeitig das Textende erreicht ist. Das Zeichen CTRL-Z wird von MS-DOS immer als Textende interpretiert.

Text an das Dateiende anfügen

Wollen Sie an das Dateiende weitere Zeilen angefügen, geschieht dies auch mit dem Befehl **I**.

Geben Sie zu diesem Befehl eine Zeilennummer ein, die größer ist als die letzte vorhandene Zeile, hängen Sie den Text immer an das Dateiende an; ebenso, wenn Sie das Numeruszeichen als Zeilennummer eingeben. Ein Beispiel:

1: Dieses ist eine Testdatei,
2: *die zeigt, wie mit dem Programm gearbeitet wird,
3: die zeigt, was passiert,
4: wenn herumgespielt wird.

An diesem Text sollen weitere Zeilen angefügt werden. Geben Sie ein:

```
#I
```

Insert

Es erscheint:

```
5:*_
```

Der neue Text wird jetzt ab der Zeile 5 an die Datei angefügt.
z.B. die Zeilen:

```
Es gibt zwei sehr brauchbare Befehle
zum Verändern der Datei:
Delete and Insert,
(der Befehl D und I),
```

Die komplette Datei sieht anschließend so aus:

```
1:  Dieses ist eine Testdatei,
2:  die zeigt, wie mit dem Programm gearbeitet wird,
3:  die zeigt, was passiert,
4:  wenn herumgespielt wird.
5:  Es gibt zwei sehr brauchbare Befehle
6:  zum Verändern der Datei:
7:  Delete and Insert,
8:*(der Befehl D und I),
```

List

Zweck

zeigt eine bestimmte Zahl von Zeilen auf dem Bildschirm

Syntax

[ZeileA] [,ZeileE] L

dabei bedeutet:

ZeileA	Anfangszeile, das ist die erste Zeile, die angezeigt werden soll.
ZeileE	Endzeile, das ist die letzte Zeile, die angezeigt werden soll.

Bemerkungen

Mit diesem Befehl listen Sie eine oder mehrere bezeichnete Zeilen der Datei auf dem Bildschirm auf. Werden mehr als 24 Zeilen ausgegeben, können Sie die Ausgabe mit der Tastenkombination **CTRL-S** vorübergehend anhalten.

Ohne Eingabe einer Zeilennummer werden 23 Zeilen aufgelistet, 11 Zeilen vor der Arbeitszeile, die Arbeitszeile und 11 Zeilen nach der Arbeitszeile.

Bei Eingabe von ZeileA und ZeileE werden diese beiden Zeilen und alle dazwischen befindlichen aufgelistet. Entfällt die erste Zeilenangabe, wird beginnend ab der elften Zeile vor der Arbeitszeile an aufgelistet. Entfällt die zweite Zeilenangabe, werden 22 Zeilen ab der eingegebenen Zeilennummer aufgelistet.

Durch die Auflistung der Zeilen auf dem Bildschirm ändern Sie nicht die Arbeitszeile (im Gegensatz zum Befehl P[age]). Bei der Auflistung wird die Arbeitszeile immer mit dem Stern (*) markiert angezeigt.

List

Auflistung bestimmter Zeilen

Für das Beispiel wird folgender Text benutzt:

1: Dieses ist eine Testdatei,
2: die zeigt, wie mit dem Programm gearbeitet wird,
3: die zeigt, was passiert,
4: wenn herumgespielt wird.
5: Es gibt zwei sehr brauchbare Befehle
6: zum Verändern der Datei:
.
.
15:*Die Arbeitszeile ist durch einen Stern markiert.
.
.
35: Dieses ist die letzte Zeile.

Wollen Sie die Zeilen 2 bis 6 auflisten lassen, geben Sie ein:

```
2,6 L
```

Es erscheint:

```
2: die zeigt, wie mit dem Programm gearbeitet wird,
3: die zeigt, was passiert,
4: wenn herumgespielt wird.
5: Es gibt zwei sehr brauchbare Befehle
6: zum Verändern der Datei:
```

Auflistung bis zur Endzeile

Geben Sie nur den Wert für *ZeileE* ein, beginnt die Befehlszeile mit einem Komma. Es werden dann die Zeilen ab der elften Zeile vor der Arbeitszeile und alle Zeilen bis zur ZeileE aufgelistet. Geben Sie ein:

,35 L

erscheinen folgende Zeilen der Beispieldatei:

```
 4:  wenn herumgespielt wird.
 5:  Es gibt zwei sehr brauchbare Befehle
 6:  zum Verändern der Datei:
 .
 .
15:*Die Arbeitszeile ist durch einen Stern markiert.
 .
 .
35:  Dieses ist die letzte Zeile.
```

Auflistung ab der eingegebenen Zeile

Geben Sie nur den Wert für *ZeileA* ein, werden dann ab der eingegebenen Zeilennummer die folgenden 23 Zeilen aufgelistet (maximal bis zum Dateiende). In diesen Zeilen muß die Arbeitszeile nicht enthalten sein. Geben Sie für die Beispieldatei ein:

1 L

werden die ersten 23 Zeilen der Datei angezeigt:

```
 1:  Dieses ist eine Testdatei,
 2:  die zeigt, wie mit dem Programm gearbeitet wird,
 3:  die zeigt, was passiert,
 4:  wenn herumgespielt wird.
 5:  Es gibt zwei sehr brauchbare Befehle
 6:  zum Verändern der Datei:
 .
 .
15:  Die Arbeitszeile ist durch einen Stern markiert.
 .
 .
23:  Die Bespieldatei besteht aus 35 Zeilen.
```

Move

Zweck

verschiebt einen Zeilenblock innerhalb der Datei

Syntax

[ZeileA] , [ZeileE] , ZeileN M

dabei bedeutet:

ZeileA	Anfangszeile, das ist die erste Zeile, die verschoben wird.
ZeileE	Endzeile, das ist die letzte Zeile, die verschoben wird.
ZeileN	Zeilennummer, das ist die Zeile vor der die verschobenen Zeilen in den Text eingefügt werden.

Bemerkungen

Mit diesem Befehl werden mehrere Zeilen innerhalb der Datei verschoben. Die Datei wird danach automatisch wieder neu numeriert.

Der Block wird ab der ZeileN wieder in die Datei eingefügt. Geben Sie den Wert ZeileA oder ZeileE nicht ein, wird dafür immer die Arbeitszeile angenommen.

Für das Beispiel wird folgender Text benutzt:

1: Dieses ist eine Testdatei,
2: die zeigt, wie mit dem Programm gearbeitet wird,
3: die zeigt, was passiert,
4: wenn herumgespielt wird.
5: Es gibt zwei sehr brauchbare Befehle
6: zum Verändern der Datei:
7:*Delete and Insert,
8: (der Befehl D und I),
9: (Der Befehl CTRL-BREAK beendet den Modus Einfügen),
10: ändern Zeilennummern.
11: Der Befehl Insert kann irgendwo in der Datei
12: Zeilen einfügen; es gibt keine
13. Probleme bei der Numerierung,
14. da diese dynamisch bis 65533 geführt wird.

Sie wollen die Zeilen 11 bis 14 vor die Zeile 5 setzen. Geben Sie dafür ein:

11,14,5 M

Die Datei sieht dann folgendermaßen aus:

1: Dieses ist eine Testdatei,
2: die zeigt, wie mit dem Programm gearbeitet wird,
3: die zeigt, was passiert,
4: wenn herumgespielt wird.
5: Der Befehl Insert kann irgendwo in der Datei
6: Zeilen einfügen; es gibt keine
7. Probleme bei der Numerierung,
8. da diese dynamisch bis 65533 geführt wird.
9:*Es gibt zwei sehr brauchbare Befehle
10: zum Verändern der Datei:
11: Delete and Insert,
12: (der Befehl D und I),
13: (Der Befehl CTRL-BREAK beendet den Modus Einfügen),
14: ändern Zeilennummern.

Move

Die neue Arbeitszeile ist die Zeile, vor der der Text eingefügt worden ist. In diesem Beispiel wurden vor der alten Zeile Nr. 5 die verschobenen Zeilen eingefügt. Die Zeile 5 ist jetzt nach der automatischen Neunumerierung die Zeile 9 (gekennzeichnet durch den Stern).

Geben Sie überschneidende Zeilennummern ein, kann der Befehl nicht ausgeführt werden. Es erscheint die Fehlermeldung:

Entry error	Eingabefehler

Page

Zweck

listet die Zeilen der Datei seitenweise auf

Syntax

[ZeileA], [ZeileE] P

dabei bedeutet:

ZeileA	Anfangszeile, das ist die erste Zeile, die angezeigt wird.
ZeileE	Endzeile, das ist die letzte Zeile, die angezeigt wird.

Bemerkungen

Mit diesem Befehl werden eine oder mehrere bezeichnete Zeilen der Datei auf dem Bildschirm aufgelistet. Dabei wird immer die letzte angezeigte Zeile zur Arbeitszeile. Werden mehr als 24 Zeilen ausgegeben, können Sie die Ausgabe mit der Tastenkombination **CTRL-S** unterbrechen.

Bei Eingabe der Werte für ZeileA und ZeileE, werden diese beiden Zeilen und alle dazwischen befindlichen aufgelistet.

Entfällt die erste Zeilenangabe, wird beginnend ab der Arbeitszeile aufgelistet. Hat die Arbeitszeile eine größere Nummer als die eingegebene Endzeilennummer, erfolgt keine Ausgabe.

Entfällt die zweite Zeilenangabe, werden von der bezeichneten Zeile an 23 Zeilen angezeigt. Die letzte Zeile wird wieder zur Arbeitszeile.

Geben Sie den Befehl ohne Zeilenangaben ein, werden 24 Zeilen angezeigt. Die Ausgabe beginnt dann immer mit der letzten Arbeitszeile.

Nach Auflistung der Zeilen auf dem Bildschirm wird die letzte angezeigte Zeile zur Arbeitszeile (im Gegensatz zum Befehl L[ist]).

Page

Für das Beispiel wird folgender Text benutzt:

1: Dieses ist eine Testdatei,
2: die zeigt, wie mit dem Programm gearbeitet wird,
3: die zeigt, was passiert,
4: wenn herumgespielt wird.
5: Es gibt zwei sehr brauchbare Befehle
6: zum Verändern der Datei:
.
.
15:*Die Arbeitszeile ist durch einen Stern markiert.
.
.
24: Diese Zeile ist jetzt die neue Arbeitszeile (siehe Stern).
35: Dieses ist die letzte Zeile.

Lassen Sie sich jetzt die ersten 24 Zeilen der Datei anzeigen, geben Sie ein:

```
1,24 P
```

Es erscheint folgender Text auf Ihrem Bildschirm:

```
 1: Dieses ist eine Testdatei,
 2: die zeigt, wie mit dem Programm gearbeitet wird,
 3: die zeigt, was passiert,
 4: wenn herumgespielt wird.
 5: Es gibt zwei sehr brauchbare Befehle
 6: zum Verändern der Datei:
 .
 .
15: Die Arbeitszeile ist durch einen Stern markiert.
 .
 .
24: *Diese Zeile ist jetzt die neue Arbeitszeile (siehe Stern).
```

Quit

Zweck

beendet das Programm, ohne die neue Datei zu sichern

Syntax

Q

Bemerkungen

Beenden Sie das Programm mit dem Befehl Q, kehren Sie sofort zum Betriebssystem zurück. Die gerade eingegebenen Zeilen werden nicht auf Platte gespeichert, die alte Datei wird nicht mit dem Zusatz .BAK versehen. Haben Sie einen neuen Dateinamen eingegeben, wird diese Datei nicht angelegt.

Nach Eingabe des Befehls erscheint die Frage:

Abort edit (Y/N)? Bearbeitung abbrechen (J/N)?

Beantworten Sie diese Frage mit **N**(ein) wird der Befehl ignoriert, und Sie können mit der Bearbeitung fortfahren. Geben Sie ein **Y** für Ja ein, sind Sie wieder im Betriebssystem MS-DOS.

Replace

Zweck

ersetzt in den bezeichneten Zeilen Zeichenkette1 durch Zeichenkette2

Syntax

[ZeileA] [,ZeileE] [?] R Zeichenkette1 F6 Zeichenkette2

dabei bedeutet:

ZeileA	Anfangszeile, das ist die erste Zeile, ab der ersetzt wird.
ZeileE	Endzeile, das ist die letzte Zeile, bis zu der ersetzt wird.
?	Fragezeichen, weist das Programm an, vor jedem Ersetzen die Zeile zu zeigen und zu fragen, ob ersetzt werden soll.
Zeichenkette1	bezeichnet den zu ersetzenden Text.
Zeichenkette2	bezeichnet den neuen Text.

Bemerkungen

Mit diesem Befehl können Sie in der Datei einen bestimmten Text oder Steuerzeichen suchen und ersetzen. Die Datei wird zwischen den bezeichneten Zeilen nach dem Text der Zeichenkette1 abgesucht. Der gefundene Text wird dann durch die Zeichenkette2 ersetzt. Beide Zeichenketten müssen durch das Zeichen CTRL-Z getrennt werden. Es dürfen keine Leerfelder zum Trennen benutzt werden. Geben Sie für eine Zeichenkette keinen Text ein, wird vom Programm der letzte noch gespeicherte Text angenommen. Ist kein Text gespeichert, kann der Befehl nicht ausgeführt werden.

Dieser Befehl kann auch *Kontrollzeichen* in den Text einfügen bzw. wieder entfernen. Benutzen Sie dafür die Tastenkombination CTRL-V. Damit teilen Sie dem Programm mit, daß der nächste Großbuchstabe, der folgt, ein Kontrollzeichen ist, z.B. ein CTRL-Z oder ein CTRL-C.

Jede Zeile, die im Text ersetzt wird, zeigt das Programm auf dem Bildschirm an. Geben Sie in der Befehlszeile das Fragezeichen mit ein, wird vor jedem Tausch die Zeile angezeigt, und Sie werden gefragt:

O.K.?

Nur nach der Eingabe von **Y** für Ja, wird der Text ersetzt. Betätigen Sie irgendeine andere Taste, wird die Suche fortgesetzt. Ist der zu ersetzende Text mehrmals in einer Zeile enthalten, erscheint die Frage jedesmal neu. Ist der Text komplett durchsucht, meldet sich das Programm wieder mit dem Prompt.

Geben Sie den Wert für ZeileA nicht ein, ist dies die erste Zeile nach der Arbeitszeile.

Geben Sie den Wert für ZeileE nicht ein, wird die Datei bis zum Ende abgesucht.

Für das Beispiel wird folgender Text benutzt:

1: Dieses ist eine Testdatei,
2: die zeigt, wie mit dem Programm gearbeitet wird,
3: die zeigt, was passiert,
4: wenn herumgespielt wird.
5: Es gibt zwei sehr brauchbare Befehle
6: zum Verändern der Datei:
7:*Delete and Insert,
8: (der Befehl D und I),
9: (Der Befehl CTRL-BREAK beendet den Modus Einfügen),
10: ändern Zeilennummern.
11: Der Befehl Insert kann irgendwo in der Datei
12: Zeilen einfügen; es gibt keine
13. Probleme bei der Numerierung,
14. da diese dynamisch bis 65533 geführt wird.

Replace

Ersetzen Sie im gesamten Text das Wort Datei durch das Wort DATEI, geben Sie ein:

1 RDateiF6DATEI

Da bei diesem Beispiel das Fragezeichen nicht mit eingegeben wird, ersetzt das Programm den Text automatisch. Es werden nur die Zeilen nach dem Ersetzen angezeigt:

6: zum Verändern der DATEI:
11: Der Befehl Insert kann irgendwo in der DATEI

Soll nur an bestimmten Stellen der Text ersetzt werden, geben Sie das Fragezeichen mit ein. Ein Beispiel:

1,10?Rei^ZEI

Jetzt wird in den Zeilen 1 bis 10 nach der Buchstabenkombination ei gesucht. Der Text wird ersetzt und anschließend gezeigt. Geben Sie auf die Frage ein **Y** für Ja ein, bleibt die gezeigte Zeile. Betätigen Sie eine andere Taste, wird der Text nicht ersetzt und die alte Zeile bleibt erhalten. Es wird folgendes angezeigt:

1: Dieses ist EIne Testdatei,
O.K.?

Geben Sie ein **Y** ein, um die neue Zeile zu bestätigen.

1: Dies ist EIne TestdatEI,
O.K.?

Geben Sie ein **N** ein, damit der Text nicht geändert wird.

2: die zEIgt, wie mit dem Programm gearbeitet wird,
O.K.?

Geben Sie nochmals ein **N** ein, damit der Text nicht geändert wird.

2: die zeigt, wie mit dem Programm gearbEItet wird,
O.K.?

Geben Sie ein **Y** für Ja ein, bleibt die Änderung. Es erscheint dann das Prompt und zeigt Ihnen damit an, daß der Befehl beendet ist.

Search

Zweck

sucht in den bezeichneten Zeilen die Zeichenkette

Syntax

[ZeileA] [,ZeileE] [?] S Zeichenkette

dabei bedeutet:

ZeileA	Anfangszeile, das ist die erste Zeile, ab der gesucht wird.
ZeileE	Endzeile, das ist die letzte Zeile, bis zu der gesucht wird.
?	Fragezeichen, weist das Programm an, die Zeile mit dem zu suchenden Text anzuzeigen.
Zeichen-kette	bezeichnet den zu suchenden Text.

Bemerkungen

Mit diesem Befehl können Sie in der Datei nach einem bestimmten Text suchen. Die Datei wird zwischen den eingegebenen Zeilen nach diesem Text abgesucht. Fehlt eine der beiden Zeilenangaben, wird dafür der Wert der Arbeitszeile eingesetzt. Die Zeile, die den gesuchten Text enthält, wird angezeigt und zur Arbeitszeile bestimmt. Damit ist der Befehl beendet.

Geben Sie nach den Zeilenangaben ein Fragezeichen (?) ein, wird die Zeile mit dem gesuchten Text gezeigt, und Sie werden gefragt:

O.K.?

Search

Geben Sie jetzt ein **Y** für Ja ein, ist der Befehl beendet und die gezeigte Zeile wird zur Arbeitszeile. Betätigen Sie irgendeine andere Taste, wird die Suche bis zur nächsten Zeile mit dem gesuchten Text fortgesetzt. Auch hier wird die Suche unterbrochen, und Sie werden gefragt:

O.K.?.

Geben Sie auch hier kein **Y** ein, wird die Suche wieder fortgesetzt, bis zum Ende des bezeichneten Bereiches.

Kann der von Ihnen bezeichnete Text nicht gefunden werden, erscheint die Meldung:

Not found   Nicht gefunden

Geben Sie in der Befehlszeile keinen Text ein, wird auf den zuletzt eingegebenen Suchtext zurückgegriffen. Ist dieser nicht vorhanden, wird der Befehl nicht ausgeführt.

Einmalige Suche eines Textes

Für das Beispiel wird folgender Text benutzt:

1: Dieses ist eine Testdatei,
2: die zeigt, wie mit dem Programm gearbeitet wird,
3: die zeigt, was passiert,
4: wenn herumgespielt wird.
5: Es gibt zwei sehr brauchbare Befehle
6: zum Verändern der Datei:
7: Delete and Insert,
8: (der Befehl D und I),
9: (Der Befehl CTRL-BREAK beendet den Modus Einfügen),
10: ändern Zeilennummern.
11: Der Befehl Insert kann irgendwo in der Datei
12: Zeilen einfügen; es gibt keine
13. Probleme bei der Numerierung,
14.*da diese dynamisch bis 65533 geführt wird.

Zwischen den Zeilen 2 und 8 soll nach der Buchstabenkombination *„um"* gesucht werden. Sie dürfen zwischen dem Befehl und dem Text kein Leerzeichen eingeben. Eine Eingabe ist nur erlaubt, wenn das Leerzeichen Bestandteil der Zeichenkette ist. Geben Sie ein:

2,8 Sum

Es wird folgendes angezeigt:

4: wenn herumgespielt wird.
*_

Die Buchstabenkombination *„um"* ist hier im Wort *„herumgespielt"* vorhanden. Der Befehl ist hier beendet und die Zeile 4 die neue Arbeitszeile. Soll weiter im Text gesucht werden, müssen Sie den Befehl neu eingeben (mit anderen Zeilenangaben) oder das Fragezeichen mit eingeben.

Mehrmalige Suche eines Textes

Wollen Sie sämtliche Zeilen des Textes sehen, in denen der gesuchte Text vorkommt, geben Sie vor dem Befehlsbuchstaben ein Fragezeichen (?) ein. Dann wird jede Zeile, die den Text enthält, angezeigt, und Sie werden gefragt:

O.K.?

Solange Sie kein **Y** für Ja eingeben, wird weitergesucht. Geben Sie den vorherigen Befehl leicht verändert noch einmal ein:

1,# ? Sum

Es wird jetzt die Zeile 4 angezeigt, und nach Eingabe eines beliebigen Zeichens wird die nächste Zeile mit dem Text gezeigt:

6: zum Verändern der Datei:
O.K.?

Antworten Sie hier ebenfalls mit einer beliebigen Taste, werden Ihnen noch die Zeilen 10 und 13 gezeigt, und zum Schluß erscheint die Meldung:

Not found Nicht gefunden

Transfer

Zweck

kopiert eine Datei in den gerade bearbeiteten Text

Syntax

[ZeileN] T Dateibez

dabei bedeutet:

ZeileN Zeilennummer, das ist die Zeile, vor der die bezeichnete Datei eingefügt wird

Dateibez Name der anzufügenden Datei mit Laufwerkbezeichnung und/oder Verzeichnisname

Bemerkungen

Mit diesem Befehl kann eine ASCII-Datei in die gerade bearbeitete Datei eingelesen werden. Die Datei wird ab der angegebenen Zeilennummer in den Text eingefügt. Geben Sie keine Zeilennummer mit an, wird ab der Arbeitszeile eingefügt.

In die gerade bearbeitete Datei soll ab der Zeile 7 die Datei mit dem Namen: NEUDAT.TXT eingefügt werden. Geben Sie ein:

7 T NEUDAT.TXT

Nach Ausführung des Befehles werden alle Zeilen der Datei neu numeriert. Die letzte eingelesene Zeile wird zur Arbeitszeile. Eine besondere Mitteilung erscheint nicht.

Write

Zweck

schreibt die Zeilen des Arbeitsspeichers auf Diskette oder Winchester zurück

Syntax

[Anzahl] W

dabei bedeutet:

Anzahl Zahl der zu schreibenden Zeilen

Bemerkungen

Ist die zu bearbeitende Datei zu groß, um auf einmal im Speicher bearbeitet zu werden, wird sie beim Programmaufruf nur soweit in den Speicher geladen, bis dieser zu 2/3 gefüllt ist.

Sie können dann den ersten Teil der Datei bearbeiten. Ist die Bearbeitung abgeschlossen, muß dieser Teil der Datei auf die Platte zurückgeschrieben werden. Erst danach können Sie weitere Zeilen nachladen.

Geben Sie eine Zeilenzahl vor dem Befehl ein, werden die benannten Zeilen auf die Platte geschrieben. Ein Beispiel:

125 W

Es werden jetzt maximal 125 Zeilen der Datei auf die Platte geschrieben.

Geben Sie keine Zeilenzahl mit ein, werden soviel Zeilen wieder auf die Platte zurückgeschrieben, bis der Speicher nur noch zu 1/4 gefüllt ist.

Immer erfolgt eine Neunumerierung des verbleibenden Textes.

Funktionstasten

Übersicht

Die *Funktionstasten* ermöglichen es Ihnen, mit geringem Aufwand die Eingabezeile zu verändern. Eine Übersicht der Tasten mit Ihren Funktionen sehen Sie in Tabelle 12.3. Die Arbeitsweise der einzelnen Tasten wird im folgenden Abschnitt erläutert.

Damit Sie die Eingabezeile bearbeiten können, müssen Sie diese Zeile erst in den *Tastatureingabespeicher* kopieren (bei der Befehlseingabe ist das immer die Eingabezeile). Sie bestimmen erst eine Arbeitszeile (mit Befehl EDIT), kopieren diese dann in den Zwischenspeicher und können dann diese Zeile bearbeiten.

Die meisten Funktionen, die mit den Tasten aufgerufen werden, geben Sie durch Betätigen einer einzelnen Taste ein. Für zwei Funktionen müssen Sie zwei Tasten betätigen.

Die Taste **INS** arbeitet als Schalter. Durch jedes Betätigen schalten Sie in den anderen Zustand.

Funktionstasten

Tabelle 12.3 Funktionstasten

TASTE	FUNKTION	BESCHREIBUNG
F1 oder -->	kopiert ein Zeichen	kopiert ein Zeichen aus dem Eingabespeicher in die neue Zeile
F2x	kopiert bis Zeichen x	kopiert alle Zeichen bis zum eingegebenen (x), aus dem Eingabespeicher in die neue Zeile
F3	kopiert alle Zeichen	kopiert alle Zeichen des Eingabespeichers in die neue Zeile
F4x	überspringt bis Zeichen x	überspringt alle Zeichen des Eingabespeichers bis zum eingegebenen
F5	neuer Eingabespeicher	schreibt die neue Zeile in den Eingabespeicher
F6	beendet Eingabe	gibt ein CTRL-Z (^Z) aus, das Zeichen für das Ende der Datei
F7	Nullzeichen	gibt ein CTRL-@ aus, das Zeichen wird mit dem Wert 00h gespeichert
BackSpace oder <--	löscht ein Zeichen	löscht ein Zeichen in der Eingabezeile und im Speicher
Ins	einfügen	schaltet um auf Einfügen
DEL	löschen	löscht ein Zeichen im Eingabespeicher, nicht in der Eingabezeile
Esc	Eingabe löschen	löscht die gesamte Eingabezeile, nicht den Eingabespeicher

Kopieren

Mit den Funktionen *„Kopieren“* können Sie auf verschiedene Weise Zeichen bei der Eingabe kopieren. Sie können ein oder mehrere Zeichen gleichzeitig kopieren. Die Kopierfunktionen werden mit einer einzelnen Taste oder auch mit zwei Tasten aufgerufen.

Kopiere ein Zeichen

Sie rufen diese Funktion mit den Tasten **F1** oder --> (Cursorsteuerungstaste nach rechts) auf. Es wird jeweils das erste Zeichen des Tastatureingabespeichers in die Eingabezeile kopiert. Danach wird das folgende Zeichen im Speicher zum ersten Zeichen. Damit können Sie durch mehrmaligen Tastendruck mehrere Zeichen hintereinander kopieren. Der Cursor wird dabei jeweils eine Stelle nach rechts gesetzt. Der Modus „Einfügen“ wird dabei automatisch ausgeschaltet. Sie haben die Zeile 1 der Datei mit dem Befehl EDIT aufgerufen, die Zeile wird auf dem Bildschirm ausgegeben. Ein Beispiel:

```
1: Dieses ist eine Testzeile.
1:*_
```

Diese Zeile ist ebenfalls in den Speicher kopiert worden. Sie können jetzt durch Betätigen der Taste **F1** oder --> das erste Zeichen der Zeile in die neue Eingabezeile kopieren:

```
1: Dieses ist eine Testzeile.
1:*D_
```

Betätigen Sie die Taste ein weiteres Mal, wird das nächste Zeichen kopiert:

```
1: Dieses ist eine Testzeile.
1:*Di_
```

Betätigen Sie die Taste noch zweimal, werden weitere zwei Zeichen kopiert:

```
1: Dieses ist eine Testzeile.
1:*Dies_
```

Kopiere bis Zeichen

Mit dieser Funktion kopieren Sie *mehrere Zeichen* des Tastatureingabespeichers in die neue Eingabezeile. Diese Funktion wird durch eine Kombination von 2 Tasten aufgerufen. Zuerst wird die Taste **F2** betätigt und dann die *alphanumerische Taste*, mit dem Zeichen der zu kopierenden Zeile, bis zu dem kopiert werden soll. Das Zeichen selbst wird nicht mitkopiert. Ist das Zeichen im Speicher nicht vorhanden oder wird nur die Taste **F2** betätigt, kann die Funktion nicht ausgeführt werden. Mit dieser Funktion schalten Sie den Modus „Einfügen“ aus. Sie haben die Zeile 1 der Datei mit dem Befehl EDIT aufgerufen, die Zeile wird auf dem Bildschirm gezeigt. Ein Beispiel:

```
1: Dieses ist eine Testzeile.
1:*_
```

Diese Zeile ist jetzt ebenfalls in den Speicher kopiert worden. Es sollen jetzt aus dieser Zeile die ersten drei Worte übernommen werden. Betätigen Sie dafür die Tasten **F2** und sofort danach die Taste mit dem Großbuchstaben **T**. Es erscheint folgendes:

```
1: Dieses ist eine Testzeile.
1:*Dieses ist eine _
```

Kopiere alle Zeichen

Mit dieser Funktion kopieren Sie *alle Zeichen*, die sich noch im Speicher befinden, in die neue Eingabezeile. Ist die neue Zeile noch leer, werden alle Zeichen kopiert. Ist die neue Zeile schon beschrieben, werden nur noch die *restlichen Zeichen* des Speichers kopiert. Mit dieser Funktion wird der Modus „Einfügen“ ausgeschaltet. Sie haben die Zeile 1 der Datei mit dem Befehl EDIT aufgerufen, die Zeile wird auf dem Bildschirm ausgegeben. Ein Beispiel:

```
1: Dieses ist eine Testzeile.
1:*_
```

Die neue Eingabezeile ist noch leer. Nach Betätigen der Taste **F3** erscheint folgendes:

```
1: Dieses ist eine Testzeile.
1:*Dieses ist eine Testzeile.
```

Funktionstasten

In die Eingabezeile sind bereits die Worte:„ Dieses war“ geschrieben worden:

1: Dieses ist eine Testzeile.
1:*Dieses war_

Damit sind die ersten 8 Zeichen des Speichers mit den neuen Zeichen überschrieben. Der Rest ist noch unverändert im Speicher. Mit der Taste **F3** kann der Rest jetzt in die Eingabezeile kopiert werden. Die Zeile sieht dann so aus:

1: Dieses ist eine Testzeile.
1:*Dieses war eine Testzeile.

Löschen

Mit den Funktionen *Löschen* können Sie auf verschiedene Weise Zeichen bei der Eingabe löschen. Sie können ein oder mehrere Zeichen gleichzeitig löschen. Ebenso können Sie gezielt nur in der *Eingabezeile* oder nur im *Tastatureingabespeicher* löschen. Die Löschfunktionen werden mit einer einzelnen Taste oder auch mit zwei Tasten aufgerufen.

Lösche ein Zeichen in der Eingabezeile

Sie rufen diese Funktion mit den Tasten **BACKSPACE** oder <-- (Cursorsteuerungstaste nach links) auf. Damit löschen Sie jeweils das letzte Zeichen der Eingabezeile. Der Tastatureingabespeicher wird dabei nicht verändert. Damit können Sie durch mehrmaligen Tastendruck mehrere Zeichen hintereinander löschen. Der Cursor wird dabei jeweils eine Stelle nach links gesetzt. Den Modus „Einfügen“ schalten Sie nicht aus. Sie haben die Zeile 1 der Datei mit dem Befehl EDIT aufgerufen und bereits teilweise neu geschrieben. Ein Beispiel:

1: Dieses ist eine Testzeile.
1:*Dieses war k_

Der zuletzt eingegebene Buchstabe „k“ wird wieder gelöscht. Danach kann der Rest der Zeile mit der Taste **F3** vom Speicher in die Eingabezeile kopiert werden:

1: Dieses ist eine Testzeile.
1:*Dieses war eine Testzeile.

Lösche ein Zeichen im Speicher

Sie rufen diese Funktion mit der Taste **DEL** auf. Es wird immer das *erste Zeichen* des Tastatureingabespeichers gelöscht. Danach wird das folgende Zeichen im Speicher zum ersten Zeichen. Damit können Sie durch mehrmaligen Tastendruck mehrere Zeichen hintereinander löschen. Der Cursor bewegt sich dabei nicht. Den Modus „Einfügen" verändern Sie dabei nicht. Sie haben die Zeile 1 der Datei mit dem Befehl EDIT aufgerufen, die Zeile wird auf dem Bildschirm ausgegeben. Ein Beispiel:

```
1: Dieses ist eine Testzeile.
1:*_
```

Durch zweimaliges Betätigen der Taste **DEL** werden die beiden ersten Zeichen des Speichers gelöscht. In der Eingabezeile wird dabei nichts angezeigt. Zur Überprüfung der Funktion können Sie mit der Taste **F3** den Rest des Speichers sichtbar machen:

```
1:  Dieses ist eine Testzeile.
1: Dieses ist eine Testzeile.
```

Lösche bis Zeichen

Mit dieser Funktion löschen Sie *mehrere Zeichen* des Tastatureingabespeichers. Diese Funktion wird durch eine Kombination von zwei Tasten aufgerufen. Zuerst wird die Taste **F4** betätigt und dann die alphanumerische Taste mit dem Zeichen, bis zu dem gelöscht werden soll. Das Zeichen selber wird nicht gelöscht. Ist das Zeichen im Speicher nicht vorhanden oder wird nur die Taste **F4** betätigt, wird nichts gelöscht. Mit dieser Funktion verändern Sie den Modus „Einfügen" nicht. Sie haben die Zeile 1 der Datei mit dem Befehl EDIT aufgerufen, die Zeile wird auf dem Bildschirm ausgegeben. Ein Beispiel:

```
1:  Dieses ist eine Testzeile.
1:*_
```

Sie wollen die beiden ersten Worte löschen. Betätigen Sie die Tasten **F4** und sofort danach die Taste mit dem Kleinbuchstaben **t**. Auf dem Bildschirm ändert sich nichts. Die Funktion können Sie wieder mit der Taste **F3** überprüfen. Es erscheint dann folgende Zeile:

```
1:  Dieses ist eine Testzeile.
1:*eine Testzeile_
```

Lösche gesamte Eingabe

Sie rufen diese Funktion mit der Taste **ESC** auf. Es wird die gesamte neu geschriebene *Eingabezeile* gelöscht. Der Eingabespeicher verändert sich dabei nicht. In die gelöschte Zeile wird ein Schrägstrich geschrieben (\) und es wird ein Zeilenvorschub ausgeführt. Die Zeile kann dann erneut eingegeben werden. Den Modus „Einfügen" schalten Sie dabei aus. Sie haben die Zeile 1 der Datei mit dem Befehl EDIT aufgerufen und bereits neu geschrieben. Auf dem Bildschirm wird folgendes angezeigt:

```
1: Dieses ist eine Testzeile.
1:*Es soll nun die Zeile_
```

Betätigen Sie jetzt erst die Taste **ESC** und dann zur Kontrolle die Taste **F3**. Es wird die Eingabe gelöscht und die alte Zeile wieder ausgegeben. Es erscheint folgendes:

```
1: Dieses ist eine Testzeile.
1:*Es soll nun die Zeile \
   Dieses ist eine Testzeile_
```

Verschiedene Funktionen

Außer den bereits erklärten Funktionen „Kopieren" und „Löschen" können Sie durch Betätigen einer einzelnen Taste noch weitere verschiedene *nützliche Funktionen* aufrufen. Sie können die Eingabezeile in den Speicher übertragen, ohne daß die Zeile in der Datei gespeichert wird. Sie können die Zeile des Speichers kopieren und beliebige Zeichen einfügen. Sie können das Dateiende markieren.

Zeichen einfügen

Sie rufen diese Funktion mit der Taste **INS** auf. Es können jetzt Zeichen in die *Eingabezeile* geschrieben werden, ohne daß gleichzeitig dafür ein Zeichen im Speicher überschrieben wird. Die einzufügenden Zeichen werden vor dem Cursor eingefügt. Die Funktion „Zeichen einfügen" bleibt solange bestehen, bis sie durch nochmali-

ges Betätigen der Taste INS oder durch eine andere Funktionstaste wieder ausgeschaltet wird. Sie haben die Zeile 1 der Datei mit dem Befehl EDIT aufgerufen und die beiden ersten Worte übernommen. Die Zeile sieht folgendermaßen aus:

```
1: Dieses ist eine Testzeile.
1:*Dieses ist_
```

Betätigen Sie die Taste **INS** und geben Sie das Wort *„jetzt"* ein:

```
1: Dieses ist eine Testzeile.
1:*Dieses ist jetzt_
```

Danach können Sie mit der Taste **F3** den Rest der Zeile kopieren. Die neue Zeile sieht so aus:

```
1: Dieses ist eine Testzeile.
1:*Dieses ist jetzt eine Testzeile._
```

Neuer Eingabespeicher

Sie rufen diese Funktion mit der Taste **F5** auf. Die neue Eingabezeile wird damit in den *Tastatureingabespeicher* kopiert, es erscheint eine neue leere Eingabezeile. Die gerade bearbeitete Zeile wird nicht in die Datei übernommen. Es wird das Zeichen @ ausgegeben und ein Zeilenvorschub ausgeführt. Danach kann diese Zeile nochmals bearbeitet werden. Sie haben die Zeile 1 der Datei mit dem Befehl EDIT aufgerufen und bereits bearbeitet:

```
1: Dieses ist eine Testzeile.
1:*Diese Zeile sollte geändert werden_
```

Sie sehen jetzt, daß diese Zeile einen Fehler enthält und wollen diesen Fehler korrigieren, bevor die Zeile gespeichert wird. Betätigen Sie die Taste **F5** und es erscheint folgendes:

```
1: Dies ist eine Testzeile.
1:*Diese Zeile sollte geändert werden@
  _
```

Datei beenden

Sie rufen diese Funktion mit der Taste **F6** auf. Es wird das Zeichen ^Z (CTRL-Z) ausgegeben. Dieses Zeichen hat den Wert 1Ah. Daran erkennt MS-DOS das Ende der Datei. Sämtliche Zeichen nach einem ^Z werden nicht wieder mit eingelesen (Ausnahme: Zusatz /B). Haben Sie im Programm EDLIN den Befehl I(nsert) aufgerufen, können Sie Ihn durch die Eingabe von ^Z beenden. Sie haben die letzte Zeile der Datei geschrieben, die nächste Zeilennummer wird angezeigt:

```
18: Dies ist die letzte Zeile.
19:*_
```

Mit Betätigen der Taste **F6** schließen Sie jetzt die Bearbeitung ab:

```
18: Dies ist die letzte Zeile.
19: ^Z
*
```

Fehlermeldungen

Sämtliche Fehlermeldungen während des Programmaufrufes führen zum sofortigen Abbruch des Programmes. Prüfen Sie dann die Eingabezeile und starten Sie das Programm erneut. Zusätzlich übergibt das Programm EDLIN den Fehlercode 3 an das Betriebssystem. Dies kann mit dem Batchbefehl IF (s.a. Kapitel 11) abgefragt werden.

Cannot edit .BAK file --rename file	Datei mit dem Zusatz .BAK kann nicht bearbeitet werden -- Datei umbenennen

ERKLÄRUNG:
Sie haben eine Datei mit dem Zusatz .BAK aufgerufen. Dieser Zusatz ist für Sicherungskopien reserviert. Wollen Sie diese Datei dennoch bearbeiten, muß die Datei umbenannt werden. Das können Sie mit dem Befehl RENAME oder mit dem Befehl COPY machen. Danach kann die Datei bearbeitet werden.

Disk Full-- Edits lost	Diskette voll-- Datei nicht komplett gesichert

ERKLÄRUNG:
Diese Fehlermeldung kann bei Eingabe des Befehls E(nd) erscheinen. Die bearbeitete Datei soll auf die Diskette geschrieben werden. Es ist nicht mehr genügend Speicherkapazität vorhanden. Der noch nicht auf Platte geschriebene Teil der Datei ist unwiderruflich verloren. Es ist jedoch noch die alte Version der Datei vorhanden, eventuell auch ein Teil der gerade bearbeiteten Datei. Achten Sie deshalb immer darauf, daß auf der Diskette für die neue Datei genügend Platz vorhanden ist.

Entry Error	Eingabefehler

ERKLÄRUNG:
EDLIN hat einen Schreibfehler in der zuletzt eingegebenen Zeile entdeckt. Überprüfen Sie die Eingabe.

Fehlermeldungen

File Creation Error — Falscher Dateiname

ERKLÄRUNG:
Der Name der Datei, die mit EDLIN bearbeitet werden soll, enthält ungültige Zeichen oder ist bereits für ein Inhaltsverzeichnis vergeben. Er kann vom Programm nicht einer einzigen Datei zugeordnet werden. Geben Sie den Namen erneut ein. Nähere Erläuterungen zum Dateinamen finden Sie im Kapitel 5.

File is READ-ONLY — Datei ist schreibgeschützt

ERKLÄRUNG:
Die aufgerufene Datei ist mit dem Programm ATTRIB geschützt worden. Die Datei kann nicht bearbeitet werden. Entfernen Sie erst mit dem Programm ATTRIB den Schreibschutz wieder.

File name must be specified — Dateiname muß angegeben werden

ERKLÄRUNG:
Sie haben das Programm EDLIN ohne Dateinamen aufgerufen. Geben Sie den Befehl erneut mit dem Namen der zu bearbeitenden Datei ein.

File not found — Datei nicht vorhanden

Sie haben mit dem Befehl T versucht, eine nicht vorhandene Datei zu lesen. Überprüfen Sie Ihre Eingabe.

Incorrect DOS-Version — Falsche DOS-Version

ERKLÄRUNG:
Sie benutzen das Programm EDLIN unter einer DOS-Version, die nicht 2.0 oder höher ist.

Invalid drive or file name	Falsches Laufwerk oder Dateiname

ERKLÄRUNG:
In der Eingabezeile ist eine falsche Laufwerksbezeichnung, ein falscher Dateiname oder ein falsches Inhaltsverzeichnis enthalten. Geben Sie den Befehl neu ein.

Invalid parameter	Falsche Zusatzeingabe

ERKLÄRUNG:
Sie haben eine nicht erlaubte Zusatzeingabe vorgenommen. Es ist nur der Zusatz /B möglich.

Line too long	Zeile zu lang

ERKLÄRUNG:
Durch den Befehl R(eplace) wurde ein längerer Text in die Zeile getauscht. Die Zeile umfaßt jetzt mehr als 253 Zeichen. Teilen Sie die Zeile auf.

Must specity destination line number	Geben Sie eine Zeilennummer ein

ERKLÄRUNG:
Sie haben den Befehl M(ove) oder C(opy) ohne Bestimmungszeile eingegeben. Überprüfen Sie Ihre Eingabe.

No room in directory for file	Kein Platz im Verzeichnis für die Datei

ERKLÄRUNG:
Sie haben den Namen einer neuen Datei eingegeben. Diese Datei kann nicht im Inhaltsverzeichnis eingetragen werden. Es ist möglich, daß das Verzeichnis voll ist. Tauschen Sie die Diskette aus. Es ist auch möglich, daß Sie einen falschen Namen oder ein falsches Laufwerk angegeben haben. Überprüfen Sie die Eingabe. Ist alles richtig und der Fehler wird wieder gemeldet, überprüfen Sie die Diskette mit dem Befehl CHKDSK.

Fehlermeldungen

Not enough room to merge the entire file — Arbeitsspeicher zu klein, die gesamte Datei kann nicht geladen werden

ERKLÄRUNG:
Mit dem Befehl T(ransfer) wird eine große Datei eingelesen. Der Arbeitsspeicher ist jedoch zu klein (bereits belegt), um die gesamte Datei auf einmal lesen zu können. Teilen Sie die Datei und wiederholen Sie dann den Befehl.

Read error in: — Lesefehler der Datei:

ERKLÄRUNG:
Sie haben mit dem Befehl T eine weitere Datei einlesen wollen. Dies ist auf Grund eines Plattenfehlers nicht möglich. Überprüfen Sie die Datei und/oder Platte mit CHKDSK.

Too many files open — Keine Dateinummer frei

ERKLÄRUNG:
Das Betriebssystem stellt für jede Datei eine Nummer zur Verfügung, mit der alle Plattenzugriffe durchgeführt werden. In der Datei CONFIG.SYS bestimmen Sie mit dem Befehl FILES die maximale Zahl. Sind alle Nummern vergeben, erscheint bei der nächsten Anforderung diese Fehlermeldung. Ändern Sie dann den Wert für FILES in der Datei CONFIG.SYS.

Kapitel 13

LIB

Übersicht

LIB ist ein MS-DOS Dienstprogramm, daß einzelne Module zu einer Bibliothek zusammenfügt. Es löscht, verändert und fügt neue Module hinzu. Das Programm finden Sie auf der zweiten Systemdiskette.

Die Arbeitsweise des Programmes gleicht der anderer Microsoftprodukte für 16-Bit-Rechner. Der Programmaufruf und die Programmanfragen sind ähnlich denen der Programme MASM, LINK, Pascal oder FORTRAN.

In diesem Kapitel wird vorausgesetzt, daß Ihnen der Maschinencode der Prozessoren 8086 und 80286 der Fa. Intel vertraut ist. Ebenso sollten Sie Begriffe wie Bit, Byte, Flag, Register und Segment kennen.

LIB erzeugt und verändert *Bibliotheken* (.LIB-Dateien), die vom Dienstprogramm LINK bearbeitet werden können. LIB kann weitere Module in diese Dateien einfügen oder einzelne Module in separate Objektmodule kopieren. Ab der Version 3 ist es mit LIB möglich, auch XENIX-Module zu verarbeiten.

Mit dem Dienstprogramm stellen Sie allgemein verwendbare Bibliotheken her, ebenso wie Bibliotheken, die sich nur auf ein spezielles Programm oder einen Compiler beziehen.

Das Programm LIB führt bei der Verwaltung der Bibliothekdatei bis zu fünf verschiedene Funktionen aus:

- Es fügt Module zu einer .LIB-Datei zusammen.
- Es löscht ein Modul.
- Es fügt ein Modul hinzu.
- Es ersetzt ein Modul.
- Es kopiert ein Modul in ein neues Objektmodul.

Arbeitsweise

Nach dem Programmaufruf werden von LIB zuerst die bezeichneten Module gelöscht. Dann werden Module herausgetrennt und zum Schluß neue Module angefügt. Dazu liest LIB jedes Modul in den Speicher ein, überprüft es auf seine Richtigkeit und schreibt es in die Bibliothek zurück (s. a. Bild 13.1). Bezeichnen Sie ein Modul zum Löschen, wird es gelesen, aber nicht zurückgeschrieben (s. a. Bild 13.2). Schreibt LIB das nächste Modul in die Datei zurück, wird dieses auf den Platz des vorherigen, gelöschten Moduls geschrieben. Damit wird Platz auf der Platte gespart (s. a. Bild 13.3).

Ist die .LIB-Datei komplett gelesen, fügt das Programm neue Module hinzu. Danach kopiert LIB Module in neue Objektmodule (s. a. Bild 13.4). Zum Schluß erstellt LIB einen Index für die Bibliothek. Je nach Größe kann dieser Vorgang bis zu 20 Sekunden dauern. Das Programm LINK benötigt diesen Index, um Module und Symbole zu finden. Haben Sie beim Programmaufruf eine Referenzliste der PUBLIC-Symbole angefordert, wird sie jetzt erstellt (s. a. Bild 13.5).

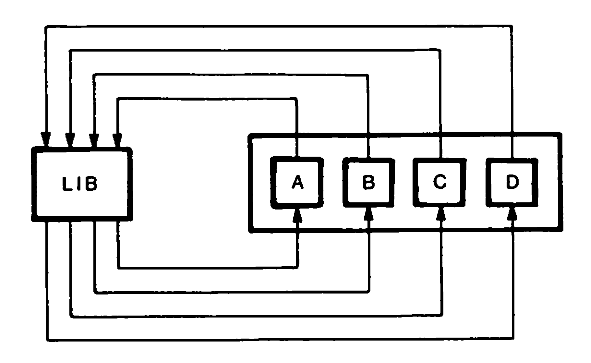


Bild 13.1
Überprüfung der Module.

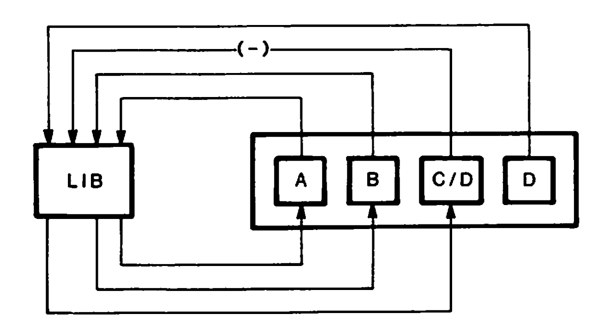


Bild 13.2
Es wird das Modul C gelöscht.
Modul D wird an den Platz von Modul C geschrieben.

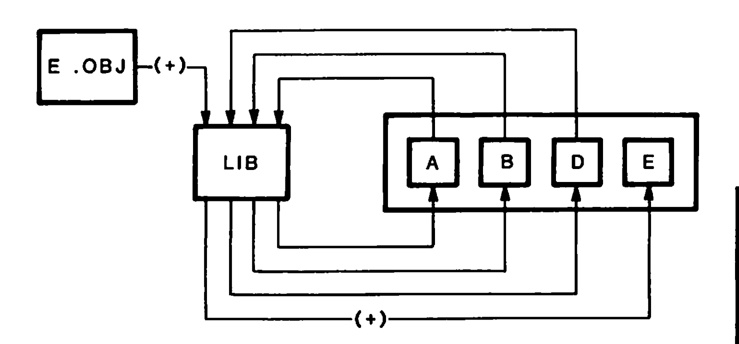


Bild 13.3
Datei E.OBJ wird als Modul E angehängt.

Arbeitsweise

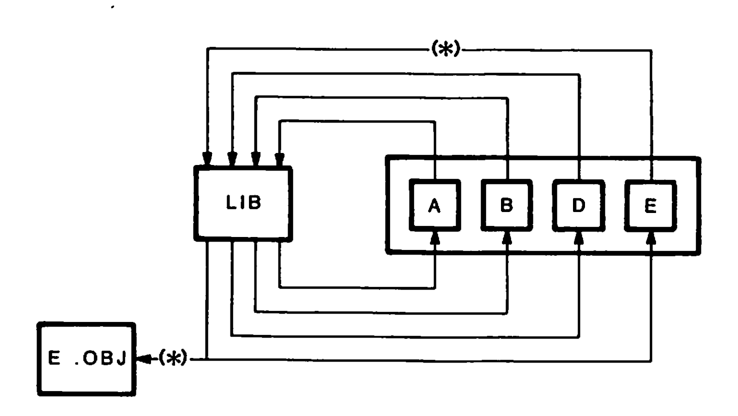


Bild 13.4
Modul E wird herausgetrennt,
in ein neues Objektmodul kopiert,
anschließend wieder in die .LIB-Datei eingefügt.

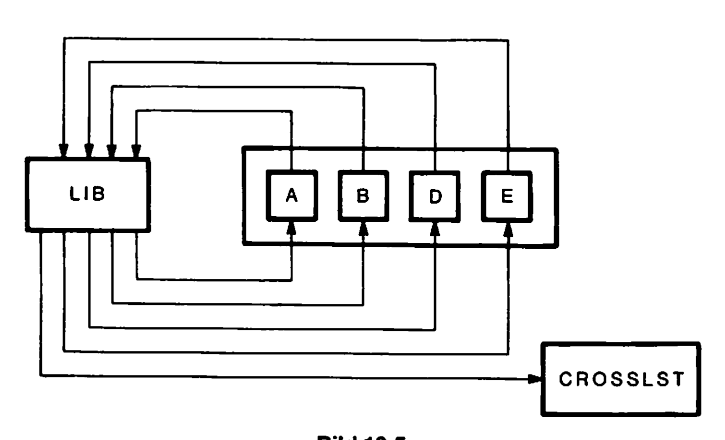


Bild 13.5
Überprüfung auf Fehler.
Erstellen einer Referenzliste aller Public-Symbole.

Ein Beispiel:

```
LIB PASCAL-HEAP+CURSOR;
```

Sie rufen das Programm LIB auf und bearbeiten die Bibliothek PASCAL. Zuerst wird das Modul HEAP.OBJ gelöscht. Danach wird das Modul CURSOR.OBJ neu hinzugefügt. Durch diese Reihenfolge der Bearbeitung wird die .LIB-Datei nicht unnötig vergrößert. Genauso wird ein Austausch von Modulen vorgenommen. Erst wird das alte Modul gelöscht, danach die neue Fassung hinzugefügt. Sie können die Bearbeitung der Module in jeder beliebigen Reihenfolge eingeben. Das Programm verfährt immer in der oben beschriebenen Weise.

Dateien von LIB

Das Programm arbeitet mit *einer Eingabedatei und erstellt zwei Ausgabedateien*. Falls erforderlich, wird eine vorübergehende Arbeitsdatei erstellt. Jede Datei wird durch ihren Namenszusatz identifiziert. Geben Sie keinen Namenszusatz an, wird vom Programm der in Tabelle 13.1 gezeigte Zusatz angenommen.

Tabelle 13.1 Dateien von LIB

ART	DATEI	ZUSATZ
Ein	Bibliothek	.LIB
Aus	Objekt	.OBJ
Aus	Liste	.LST
Ein/Aus	Arbeitsdatei	.TMP

Arbeitsweise

Arbeitsdatei VM.TMP

Das Programm LIB benutzt während des Ablaufes den gesamten Arbeitsspeicher. Wird durch die zu verbindenden Dateien eine größere Ausgabedatei erstellt, als Speicherkapazität vorhanden ist, erzeugt das Programm eine Arbeitsdatei. Diese Datei erhält den Namen *VM.TMP*. Sie dient vorübergehend als *Speichererweiterung*. In ihr werden Informationen über das neue Modul oder die neue Bibliothek abgespeichert, bevor diese in eine neue Datei geschrieben werden. Dies wird durch die Meldung

VM.TMP has been created. Do not change diskette in drive X:	Datei VM.TMP wurde angelegt Diskette in Laufwerk X: nicht wechseln

angezeigt. Erscheint diese Meldung, dürfen Sie die Diskette bis zum Ende des Programmes nicht mehr wechseln. Andernfalls wird das Programm sofort abgebrochen. Es erscheint dann folgende Meldung:

Unexpected end of file on VM.TMP	Datei VM.TMP nicht mehr vorhanden

LIB löscht die Datei am Ende des Programmablaufes wieder. Sollten Sie eine Ihrer Dateien mit diesem Namen benannt haben, überschreibt LIB diese Datei und löscht sie anschließend ebenfalls.

Programmaufruf

Syntax

LIB
LIB Bibliothek [/P:n] [Funktionen, [Listdatei, [Ausgabedatei]]]
LIB @Dateibez

dabei bedeutet:

Bibliothek Name der Datei, die bearbeitet werden soll. Dem Namen kann ein Laufwerk und/oder Verzeichnis vorangestellt werden.

/P:n Bestimmt die Größe einer Seite der Bibliothek in Bytes. n ist eine gerade Zahl im Bereich 16 - 32.768.

Funktion Befehlszeichen, bestimmt die Bearbeitung des nachfolgenden Modules.

Listdatei Dateiname für die Referenzliste. Dem Namen kann ein Laufwerk und/oder Verzeichnis vorangestellt werden.

Ausgabedatei Dateiname für die geänderte Bibliothek. Dem Namen kann ein Laufwerk und/oder Verzeichnis vorangestellt werden.

Dateibez Dateiname für die Antwortdatei. Das erste Zeichen muß immer "@" sein. Dem Namen kann ein Laufwerk und/oder Verzeichnis vorangestellt werden.

Programmaufruf

Bemerkungen

Während des Programmablaufes werden von Ihnen zwei verschiedene Eingaben gefordert:

- Programmaufruf
- Antwort auf eine Programmanfrage

Sie können dabei wählen, ob Sie alle Eingaben über die Tastatur vornehmen, oder ob Sie alle Eingaben in einer separaten Datei speichern und die Datei vom Programm abfragen lassen. Dabei können Sie bis zu sechs verschiedene Zeichen eingeben, die von LIB als Befehlszeichen interpretiert werden und den Programmablauf beeinflussen.

Programmaufruf

Sie haben drei verschiedene Möglichkeiten, das Programm aufzurufen.

Sie antworten der Reihe nach auf alle vom Programm gestellten Anforderungen; sie schreiben alle Antworten in die Befehlszeile, mit der das Programm gestartet wird; sie schreiben alle Antworten in eine Datei, die Sie vorher erstellen.

Tabelle 13.2 Übersicht LIB - Programmaufruf

ART	EINGABE
Einzelne Antwort	LIB
Antwort in der Befehlszeile	LIB Bibliothek [/P:n] Funktionen, [Listdatei], [Ausgabedatei]
Antwortdatei	LIB @Dateiname

Einzelne Antwort

Wollen Sie das Programm nur mit seinem Namen aufrufen, geben Sie in der Befehlszeile ein:

```
LIB
```

Das Programm wird in den Speicher geladen. Die zur Bearbeitung benötigten Dateien werden jetzt einzeln durch das Programm abgefragt. Mit der Antwort auf die folgenden Anfragen bestimmen Sie die Aufgaben des Programmes. Eine Übersicht zeigen die beiden Tabellen 13.2 und 13.3. Ausführliche Erklärungen folgen in den weiteren Abschnitten des Kapitels.

Tabelle 13.3 LIB-Antworten

FRAGE	ANTWORT
Bibliothek	Eingabe des Namens der Datei, die bearbeitet werden soll. Dem Namen kann ein Laufwerk und/oder Verzeichnis vorangestellt werden. Geben Sie keinen Namenszusatz ein, wird .LIB angenommen.
Funktion	Eingabe eines Befehlszeichens, bestimmt die Bearbeitung des nachfolgenden Moduls. Geben Sie nur den Modulnamen ein, erfolgt keine Veränderung und als Namenszusatz des Moduls wird .OBJ angenommen.
Listdatei	Eingabe des Dateinamens für die Referenzliste. Ohne Eingabe wird keine Liste erstellt.
Ausgabedatei	Eingabe eines Namens für die geänderte LIB-Datei. Dem Namen kann ein Laufwerk und/oder Verzeichnis vorangestellt werden. Die alte LIB-Datei erhält den Zusatz BAK. Ohne Eingabe werden alle Änderungen in einer neuen Datei mit dem bisherigen Namen vorgenommen.

Programmaufruf

Tabelle 13.4 LIB-Befehlszeichen

ZEICHEN	FUNKTION
+	Fügt das Modul an die Datei an.
-	Löscht das Modul der Datei.
*	Kopiert das Modul in eine eigene Objektdatei.
;	Übernimmt die vorgegebenen Werte für die restlichen Fragen.
&	Verlängert die Arbeitszeile in die nächste physikalische Zeile.
CTRL-BREAK	Bricht die Bearbeitung ab.

Das Zeichen „*" kann mit den Zeichen „+" oder „-" kombiniert werden.

Antwort in der Befehlszeile

Rufen Sie das Programm gleichzeitig mit allen zu bearbeitenden Namen auf, geben Sie folgende Befehlszeile ein:

```
LIB Bibliothek [/P:n] Funktion, [Listdatei], [Ausgabedatei]
```

Sämtliche Eingaben nach dem Programmnamen LIB werden als Antwort auf die Anfragen gewertet. Die einzelnen Eingabefelder müssen durch ein Befehlszeichen getrennt werden. Soll eine Referenzliste erstellt werden, geben Sie den Namen der Listdatei, durch ein Komma getrennt, ein. Dies gilt auch für eine neue LIB-Datei.

Die Bedeutung der einzelnen Felder der Befehlszeile:

Bibliothek Eingabe des Namens der .LIB-Datei, die bearbeitet werden soll. Es ist möglich, dem Namen ein Laufwerk und/oder Verzeichnis voranzustellen. Wird kein Namenszusatz eingegeben, wird .LIB angenommen. Ist der Name nicht vorhanden, werden Sie gefragt:

Library does not exist. Create?	Bibliothek ist nicht vorhanden. Anlegen?

Geben Sie ein **Y** für Ja ein, wird eine neue Datei angelegt. Jede andere Eingabe bricht das Programm ab und bringt Sie zum Betriebssystem zurück.

/P:n Der Wert für n bestimmt die Größe einer Seite in der neuen LIB-Datei. Damit bestimmen Sie die Startadresse eines neuen Modules in der Datei. n muß eine gerade Zahl im Bereich 16 - 32.768 sein. Bestimmen Sie eine größere Seite, kann die Datei mehr Module speichern. Ohne Eingabe von /P beträgt die Größe einer Seite 16 Bytes.

Funktion Eingabe der Modulnamen mit einem der folgenden Be-fehlszeichen zum Bestimmen der Funktion. Das Pluszeichen (+) fügt das Modul der Datei an. Das Minuszeichen (-) löscht das Modul und der Stern (*) kopiert das Modul in eine neue Objektdatei. Vor dem Modulnamen darf ein Laufwerk und/oder ein Verzeichnis eingegeben werden.

Listdatei Eingabe des Dateinamens für die Referenzliste. Die Liste enthält eine Aufzählung der PUBLIC-Symbole. Sie wird zum Schluß der Bearbeitung erstellt.

Ausgabedatei Eingabe eines Namens für die geänderte LIB-Datei. Dem Namen kann ein Laufwerk und/oder Verzeichnis vorangestellt werden. Ohne Eingabe werden alle Änderungen in eine neue Datei mit dem bisherigen Namen geschrieben. Die alte LIB-Datei erhält den Zusatz BAK.

Programmaufruf

Geben Sie nach dem Namen der .LIB-Datei nur ein Semikolon (;) ein, wird diese Datei in den Speicher geladen. Die Module werden auf Fehler überprüft. Änderungen werden nicht durchgeführt. Ein Beispiel:

```
LIB PASCAL;
```

Die Bibliothek PASCAL wird auf Fehler überprüft, ohne daß weitere Änderungen vorgenommen werden.

Geben Sie nach dem Namen der .LIB-Datei ein Komma mit dem Namen der Listdatei ein, wird die Datei auf Fehler überprüft und anschließend die Referenzliste erstellt. Ein Beispiel:

```
LIB PASCAL,PASCROSS.LST
```

LIB überprüft die Bibliothek PASCAL auf Fehler und erstellt anschließend eine Referenzliste mit dem Namen PASCROSS.LST.

Geben Sie nach dem Namen der LIB-Datei mehrere Modulnamen ein, muß jedem Modul ein Befehlszeichen vorangestellt werden. Erstreckt sich die Eingabe über mehr als eine Zeile, muß das letzte Zeichen der Zeile ein „Und"-Zeichen (&) sein. Ein Beispiel:

```
LIB PASCAL*CLS+HEAP&
-DOWN.EXT,,PASNEU
```

LIB lädt die Bibliothek PASCAL. Zuerst wird das Modul DOWN.EXT gelöscht, dann wird das Modul CLS.OBJ in eine neue Objektdatei kopiert. Zum Schluß wird das Modul HEAP.OBJ angefügt. Eine Referenzliste wird nicht erstellt. Die geänderte LIB-Datei erhält den Namen PASNEU.LIB.

Antwortdatei

Ihre Antworten auf die Befehlsanfragen von LIB können Sie in einer separaten *Textdatei* abspeichern. Das erste Zeichen der Datei muß immer „ " sein. Nur daran erkennt LIB, daß es die Antwortdatei ist. Rufen Sie das Programm dann folgendermaßen auf:

```
LINK @Dateibez
```

Dem Namen der Antwortdatei kann ein Laufwerk und/oder der Name eines Verzeichnisses vorangestellt werden. Sie können diese Datei mit dem Programm EDLIN erstellen oder auch direkt von der Tastatur aus eingeben.

Die Antwortdatei ist eine Textdatei. Sie muß alle Antworten in der korrekten Reihenfolge enthalten. Jede Antwort muß in einer eigenen Zeile stehen. Längere Antworten können über mehrere Zeilen eingegeben werden, wenn diese Zeilen durch das „Und"-Zeichen (&) verbunden werden.

Nach dem Programmaufruf werden sämtliche Fragen und Antworten zur Kontrolle auf dem Bildschirm angezeigt. Enthält die Datei auf eine Frage keine Antwort, wird der vorgegebene Wert angenommen. Ein Beispiel:

PASCAL
+CURSOR+HEAP-CLS*DOWN
CROSSLST.MAP
PASNEU

In diesem Beispiel rufen Sie die Datei PASCAL.LIB auf. Zuerst löschen Sie das Modul CLS.OBJ, danach kopiert LIB das Modul DOWN.OBJ in eine separate Objektdatei. Zuletzt werden die Module CURSOR.OBJ und HEAP.OBJ am Ende der Bibliothek angefügt. Die Referenzliste erhält den Namen CROSSLIST.MAP. Die geänderte LIB-Datei heißt PASNEU.LIB, die bisherige Datei wird von PASCAL.LIB in PASCAL.BAK umbenannt.

Programmablauf

Nach dem Start von LIB werden Sie aufgefordert, vier verschiedene Dateinamen einzugeben. Die *Abfrage* erfolgt schrittweise. Nach der letzten Eingabe beginnt das Programm mit der Ausführung. Tritt kein Fehler auf, wird das Programm normal beendet, und Sie sind wieder im Betriebssystem. Findet das Programm einen Fehler, erscheint die entsprechende Fehlermeldung und das Programm wird abgebrochen. Nach der Behebung des Fehlers müssen Sie LIB neu starten.

Bei der Anfrage der Dateinamen handelt es sich um die zu bearbeitende Bibliothek, die zu bearbeitenden Module, die Referenzliste der PUBLIC-Symbole und die neue LIB-Datei. Zusätzlich können Sie in der Befehlszeile mit der Eingabe für PAGESIZE die Größe einer Seite in der neuen LIB-Datei bestimmen.

Bibliothek

Geben Sie hier den Namen der *.LIB-Datei* ein, die bearbeitet werden soll. Dem Namen kann ein Laufwerk und/oder Verzeichnis vorangestellt werden. Der vorgegebene Namenszusatz ist *.LIB*. Sie können ihn durch Eingabe eines anderen Zusatzes überschreiben. Nach dem Namen dürfen Sie außer einem Semikolon nur den Zusatz /P und/oder ein Semikolon eingegeben. Das Semikolon bedeutet, daß die Datei nur überprüft wird. Geben Sie andere Zeichen ein, werden diese ignoriert. Ist der Name der Bibliothek nicht vorhanden, werden Sie gefragt:

Library does not exist. Create?	Bibliothek ist nicht vorhanden. Anlegen?

Geben Sie ein Y für Ja ein, wird eine neue Datei angelegt. Jede andere Eingabe bricht das Programm ab und bringt Sie zum Betriebssystem zurück.

/PAGESIZE:n

Mit dem Wert für *n* bestimmen Sie die Größe einer Seite in der neuen LIB-Datei. Es dürfen nur geradzahlige Werte im Bereich 16 bis 32.768 eingegeben werden. Der Zusatz muß immer durch einen Schrägstrich von den vorherigen Eingaben der Befehlszeile getrennt werden. Das Wort PAGESIZE darf auch mit dem Buchstaben **P** abgekürzt werden.

Das Programm LIB speichert alle *PUBLIC-Symbole* in einem Index zu Beginn der Datei. Daher kann eine Datei mit einer größeren Seite eine größere Anzahl von Symbolen und Modulen speichern. Jedes neue Modul beginnt immer zu Beginn einer neuen Seite. Damit werden aber auch durchschnittlich *n*/2 Bytes pro Modul Speicherplatz nicht benutzt. Deshalb sollten Sie den Wert für *n* nur so groß wie nötig berechnen. Ein Beispiel:

```
LIB BAS /P:256;
```

Mit dieser Eingabe wird der Datei BAS.LIB eine Seitengröße von 256 Bytes zugeteilt.

Funktionen

Geben Sie hier eines der drei *Befehlszeichen* ein, mit dem die Art der Bearbeitung für das folgende Modul bestimmt wird. Nach dem Zeichen muß der Name des Moduls (falls erforderlich mit Laufwerknamen und/oder vorangestelltem Verzeichnis) ohne Leerzeichen eingegeben werden. Das Pluszeichen (+) fügt das Modul an das Ende der Datei an. Das Minuszeichen (-) löscht das Modul in der Datei. Die Kombination „ -+ " löscht das Modul der LIB-Datei und ersetzt es durch ein gleichnamiges OBJ-Modul. Der Stern (*) kopiert das Modul in eine neue Objektdatei. Die Kombination „ -* " löscht das Modul der LIB-Datei und kopiert es in eine eigene OBJ-Datei. Die Datei wird dabei nach dem Modul benannt und mit dem Zusatz .OBJ versehen.

Umfaßt Ihre Eingabezeile mehr als 80 Zeichen, geben Sie als letztes Zeichen der Zeile ein „Und"-Zeichen (&) ein. Die Befehlsanfrage wird dann in der nächsten Zeile wiederholt und wie eine einzige Eingabe behandelt. Die zu bearbeitenden Module können in jeder beliebigen Reihenfolge eingegeben werden. Die Module bearbeitet LIB immer in der Reihenfolge: Löschen, Ersetzen, Anfügen.

Listdatei

Wünschen Sie eine Übersicht sämtlicher Module und der darin enthaltenen *PUBLIC-Symbole*, geben Sie einen Namen für die Listdatei ein. Dieser Name muß durch ein Komma von der übrigen Zeile getrennt werden. Die Datei wird zum Schluß der Bearbeitung erstellt und enthält zwei Listen. In der ersten Liste sind alle PUBLIC-Symbole alphabetisch mit Namen des Moduls aufgeführt. Die zweite Liste enthält in alphabetischer Reihenfolge die Namen der Module mit den darin enthaltenen Symbolen. Geben Sie keinen Namen ein, wird die Liste nicht erstellt. Soll der Dateiname einen Zusatz enthalten, müssen Sie auch diesen mit eingeben. Dem Namen darf ein Laufwerk und/oder ein Verzeichnis vorangestellt werden.

Ausgabedatei

Geben Sie hier eine Dateibezeichnung ein für die neue, überarbeitete *LIB-Datei*. Dem Dateinamen kann außer einem Laufwerk auch ein Verzeichnisname vorangestellt werden. Ohne weitere Eingabe übernimmt das Programm den bisherigen Dateinamen. Die alte Datei erhält dann den Zusatz BAK. Geben Sie nur einen neuen Namen, aber keinen Zusatz ein, wird dafür vom Programm automatisch LIB eingesetzt. Schließen Sie Ihre Eingabe mit RETURN ab. Das Programm LIB beginnt dann mit der Bearbeitung.

Befehlszeichen

Es ist die Eingabe von sechs verschiedenen Zeichen in der Befehlszeile erlaubt. Drei Zeichen rufen eine *Funktion* zum Bearbeiten der Module auf, die anderen Zeichen unterstützen die Befehlseingabe.

Pluszeichen (+)

Das Modul, das dem Pluszeichen folgt, wird als letztes Modul an die gerade bearbeitete Datei *angehängt*. Schreiben Sie den Modulnamen ohne Namenszusatz, wird vom Programm .OBJ angenommen. In der .LIB.Datei selbst, wird nur der Name des Moduls benutzt. Hat das neue Modul z. B. den Namen: CURSOR.MOD und befindet sich auf Laufwerk B:, dann geben Sie auf die Anfrage ein:

```
+B:CURSOR.MOD
```

In der Bibliothek wird das Modul dann unter dem Namen CURSOR geführt. Mit dieser Funktion können auch andere Bibliotheken oder XENIX-Module in die bestehende Bibliothek aufgenommen werden.

Minuszeichen (-)

LIB *löscht* das Modul, das dem Minuszeichen folgt, aus der Datei. Alle nachfolgenden Module rücken auf, um den freien Platz in der Datei zu schließen. Dadurch wächst die Datei nicht unnötigerweise. Wechseln Sie ein Modul aus, müssen Sie es erst löschen und danach wieder neu an das Ende anhängen. Geben Sie dafür vor dem Modulnamen die Zeichenkombination „ -+ " ein. Soll das Modul aus der Bibliothek heraus in eine neue Objektdatei verschoben werden, geben Sie die Zeichenkombination „ -* " ein. Damit wird das Modul in der Bibliothek anschließend gelöscht.

Programmablauf

Stern (*)

Das Modul, das dem Stern folgt, wird in eine neue Objektdatei *kopiert.* Es ist aber weiterhin in der .LIB-Datei vorhanden. Dabei wird für das Objektmodul der bisherige Name übernommen und mit dem Zusatz .OBJ versehen. Dies können Sie anschließend ändern. Ein Beispiel:

Ist der Name des Moduls in der Datei CURSOR, dann geben Sie ein:

```
*Cursor
```

Es wird auf dem Arbeitslaufwerk das Modul CURSOR.OBJ erstellt.

Semikolon (;)

Das Semikolon dürfen Sie nur als letztes Zeichen in der Befehlszeile eingeben. Nach dem Semikolon fragt das Programm keine Dateinamen mehr ab. Es werden die *vorgegebenen Werte* angenommen. (Eine Referenzliste wird nicht erstellt.) Geben Sie das Semikolon nach dem Namen der .LIB-Datei ein, wird diese nur auf Fehler untersucht und nicht verändert. Wollen Sie ein Feld übergehen, betätigen Sie die Taste **RETURN**. Ein Beispiel:

```
Library file: FUN
Operations: +CURSOR;
```

Es wird nach dem Namen der Referenzliste nicht mehr gefragt. Die Bearbeitung beginnt sofort.

„Und"-Zeichen (&)

Mit dem „Und"-Zeichen wird die logische Zeile in der nächsten physikalischen Zeile *fortgesetzt.* Dieses Zeichen benötigen Sie nur für die Eingabe der Modulnamen (Operations:). Sie können beliebig viele Modulnamen eingeben, nur begrenzt durch den Arbeits- oder Diskettenspeicher.

Ist die Eingabe länger als 68 Zeichen, geben Sie nach einem Modulnamen das Zeichen & ein und betätigen die Taste **RETURN**. In der folgenden Zeile erscheint wieder die Systemanfrage und Sie können weitere Modulnamen eingeben. Ein Beispiel:

```
Library file: FUN
Operations: +CURSOR-HEAP+HEAP*CLS&
Operations: _
```

Nach dem letzten Modulnamen der ersten Zeile (CLS) haben Sie das Und-Zeichen eingegeben. Es erscheint in der folgenden Zeile wieder die Befehlsanfrage: Operations:. Sie können noch weitere Module benennen.

CTRL - BREAK

Mit der Tastenkombination **CTRL-BREAK** kann das Programm jederzeit *abgebrochen* werden. Geben Sie falsche Dateinamen ein, können Sie diese Namen in der Arbeitszeile mit den Funktionstasten korrigieren. Andere Namen können nicht mehr geändert werden. Brechen Sie dann das Programm ab und starten Sie neu. Auf dem Bildschirm erscheint die Meldung:

```
Status 1
```

Das Programm LIB übergibt damit gleichzeitig einen Fehlerkode an das Betriebssystem. Der Wert ist 1 und kann in einer Batchdatei mit dem Befehl ERRORLEVEL abgefragt werden.

Fehlermeldungen

Das Programm wird durch die meisten Fehlermeldungen abgebrochen. Es muß anschließend wieder neu gestartet werden. Ein Fehlerkode wird damit jedoch nicht übergeben.

Symbol is a multiply defined PUBLIC. Proceed?	Das Symbol ist mehrfach definiert. Fortfahren?

ERKLÄRUNG:
Zwei Module definieren dasselbe PUBLIC-Symbol. Benennen Sie ein Symbol um und assemblieren Sie das Modul neu.

Allocate error on VM.TMP	Fehler der Datei VM.TMP

ERKLÄRUNG:
Die Arbeitsdatei VM.TMP kann nicht angelegt werden, weil die Diskette voll ist. Wechseln Sie die Diskette und starten Sie LIB neu.

Cannot access file	Datei nicht vorhanden

ERKLÄRUNG:
Die bezeichnete OBJ-Datei ist nicht vorhanden. Überprüfen Sie Ihre Eingabe.

Cannot create extract file	Das Objektmodul kann nicht erstellt werden.

ERKLÄRUNG:
Das neue Objektmodul kann nicht angelegt werden, weil die Diskette voll ist. Wechseln Sie die Diskette und starten Sie LIB neu.

Cannot nest response file	Antwortdatei kann nicht verbunden werden.

ERKLÄRUNG:
Die von Ihnen erstellte Antwortdatei ruft eine andere Antwortdatei auf. Schreiben Sie alle Antworten in eine Datei.

Fehlermeldungen

Cannot open VM.TMP	Die Arbeitsdatei kann nicht angelegt werden

ERKLÄRUNG:
Die Datei VM.TMP, die zur Speichererweiterung benötigt wird, kann nicht angelegt werden, weil die Diskette voll ist. Nehmen Sie eine andere Diskette.

Cannot write library file	Die LIB-Datei kann nicht geschrieben werden

ERKLÄRUNG:
Die Bibliothek kann nicht auf die Diskette zurückgeschrieben werden, weil diese voll ist. Nehmen Sie eine andere Diskette.

Close error on extract file	Fehler in der neuen Objektdatei

ERKLÄRUNG:
Die neue Objektdatei kann nicht auf die Diskette geschrieben werden, weil diese voll ist. Nehmen Sie eine andere Diskette.

Error writing to cross reference file	Schreibfehler der Referenzliste

ERKLÄRUNG:
Das Programm kann die gewünschte Listdatei nicht anlegen. Die Diskette ist voll oder das Inhaltsverzeichnis kann keinen weiteren Eintrag mehr speichern. Wechseln Sie die Diskette.

Error writing to new library	Schreibfehler der neuen LIB-Datei

ERKLÄRUNG:
Das Programm kann die geänderte LIB-Datei nicht speichern. Die Diskette ist voll oder das Inhaltsverzeichnis kann keinen weiteren Eintrag mehr speichern. Wechseln Sie die Diskette.

LIB

Fehlermeldungen

Fatal Error: An internal error has occurred	Schwerer Fehler: Ein interner Fehler ist aufgetreten

ERKLÄRUNG:
Teilen sie diesen Fehler Zenith Data Systems mit. Er wird an die Fa. Microsoft weitergemeldet.

Fatal Error: Cannot create list file	Schwerer Fehler: Die Listdatei kann nicht erstellt werden

ERKLÄRUNG:
Der Name der Listdatei ist fehlerhaft oder die Diskette ist voll. Überprüfen Sie Ihre Eingabe und die Diskette.

Fatal Error: Cannot open response file	Schwerer Fehler: Die Antwortdatei kann nicht gelesen werden

ERKLÄRUNG:
Der Name der Antwortdatei ist fehlerhaft oder die Datei ist nicht auf der Diskette. Überprüfen Sie Ihre Eingabe und die Diskette.

Fatal Error: Invalid object module/library	Schwerer Fehler: Objektmodul oder .LIB-Datei falsch

ERKLÄRUNG:
Das Objektmodul oder die .LIB-Datei sind nicht vorhanden oder fehlerhaft. Ebenso ist es möglich, daß die Diskette beschädigt ist. Überprüfen Sie Ihre Eingabe und die Diskette.

Fatal Error: Module is not in the library	Schwerer Fehler: Objektmodul ist nicht in der .LIB-Datei

ERKLÄRUNG:
Es sollte ein Modul bearbeitet werden, das nicht vorhanden ist. Überprüfen Sie Ihre Eingabe.

Fehlermeldungen

Fatal Error: write error on library/extract file	Schwerer Fehler: Schreibfehler der Dateien

ERKLÄRUNG:
Die Diskette hat nicht mehr genug Speicherkapazität, um die neuen Dateien aufzunehmen. Wechseln Sie die Diskette.

Input file read error	Lesefehler der Datei

ERKLÄRUNG:
Das Modul ist fehlerhaft, eventuell ein Fehler der Diskette. Überprüfen Sie das Modul.

Incorrect DOS version, use DOS 2.0 or later	Falsche DOS-Version, benutzen Sie DOS 2.0 oder höher

ERKLÄRUNG:
Sie benutzen das Programm LIB unter einer DOS-Version, die nicht 2.0 oder höher ist. Starten Sie das System mit einer anderen DOS-Version neu.

Insufficient stack space	Arbeitsspeicher für den Stack-Bereich zu klein

ERKLÄRUNG:
Der Arbeitsspeicher ist zu klein. Das Programm LIB erhält nicht den benötigten Speicherbereich. Starten Sie das System neu, verringern Sie die Zahl der Zwischenspeicher oder zusätzlicher Gerätetreiber.

Library Disk is full	Diskette ist voll

ERKLÄRUNG:
Die Diskette hat nicht mehr genug Speicherkapazität, um die neuen Dateien aufzunehmen. Wechseln Sie die Diskette.

Fehlermeldungen

Listing file write error	Schreibfehler der Listdatei

ERKLÄRUNG:
Die Diskette hat nicht mehr genug Speicherkapazität, um die neue Datei aufzunehmen. Wechseln Sie die Diskette.

Output library specification ignored	Name für die neue LIB-Datei nicht beachtet

ERKLÄRUNG:
Sie haben in der Befehlszeile einen falschen oder fehlerhaften Namen eingegeben. Überprüfen Sie Ihre Eingabe.

No library file specified	Keine .LIB-Datei benannt

ERKLÄRUNG:
Die Frage nach der Bibliothek wurde nicht beantwortet. Starten Sie LIB neu und geben Sie einen Namen ein.

Page size too small	Seitengröße zu klein

ERKLÄRUNG:
Sie haben für den Zusatz /P einen falschen Wert eingegeben. Erlaubt ist ein Wert im Bereich 16 bis 32.768.

Read error on VM.TMP	Lesefehler der Datei VM.TMP

ERKLÄRUNG:
Die Diskette mit der Datei VM.TMP ist beschädigt. Tauschen Sie die Diskette aus.

Symbol table capacity exceeded	Speicherkapazität der Symboltabelle überschritten

ERKLÄRUNG:
Durch zu viele oder zu lange Namen ist der Speicher der Tabelle (ca. 30Kb) voll. Löschen Sie nicht mehr benötigte Module oder kopieren Sie diese in eine neue .LIB-Datei.

Fehlermeldungen

Syntax error (bad file spec)	Schreibfehler Falsche Dateibezeichnung

ERKLÄRUNG:
Sie haben in der Befehlszeile vor dem Namen der LIB-Datei den Zusatz /P eingegeben. Vertauschen Sie die Eingabe.

Too many objekt modules	Zu viele Objektmodule

ERKLÄRUNG:
Es sind maximal 500 Module erlaubt. Löschen Sie nicht mehr benötigte Module oder kopieren Sie diese in eine neue .LIB-Datei.

Too many PUBLIC symbols	Zu viele PUBLIC-Symbole

ERKLÄRUNG:
Es sind maximal 1024 Symbole erlaubt. Ändern Sie die Objektmodule und starten Sie LIB neu.

Unknown switch	Falscher Zusatz

ERKLÄRUNG:
Sie haben einen falschen Zusatz eingegeben. Es ist nur der Zusatz /P für PAGESIZE erlaubt.

Write error on VM.TMP	Schreibfehler der Datei VM.TMP

ERKLÄRUNG:
Die Speicherkapazität der Diskette reicht nicht aus, um weitere Daten der Datei VM.TMP aufzunehmen. Wechseln Sie die Diskette und starten Sie LIB neu.

Kapitel 14

LINK

Übersicht

LINK ist ein MS-DOS-Dienstprogramm, daß einzelne Dateien, die im 8086- oder 80286-Objekt-Code erstellt wurden, *miteinander verbindet*. Das Programm ist auf der zweiten Systemdiskette enthalten.

In diesem Kapitel wird vorausgesetzt, daß dem Leser der Maschinencode der Prozessoren 8086 und 80286 der Fa. Intel vertraut ist. Ebenso sollten Begriffe wie Bit, Byte, Flag, Register und Segment bekannt sein.

Das Programm LINK verbindet mehrere Module, die im Objektkode der Prozessoren erstellt sein müssen. Die so erstellte Datei kann dann ohne Beachtung von Speicheradressen in den Arbeitsspeicher geladen werden. Das Programm kann einzelne Module bis zu einer Gesamtgröße von 1 MB verbinden. Das so erstellte neue Programm ist dann sofort ablauffähig. Zum Absuchen der LIB-Dateien benutzt LINK eine bibliothekindexierte Suchmethode.

Zusätzlich löst LINK externe Querreferenzen zwischen den einzelnen Objektmodulen. Das Programm kann maximal sechzehn LIB-Dateien nach Definitionen von unbekannten Referenzen absuchen. Ebenso kann das Programm ein Listing der gelösten externen Referenzen und der gefundenen Fehler erstellen.

Dabei wird der gesamte Arbeitsspeicher benutzt. Ist der Speicher zu klein, wird vom Programm automatisch eine vorübergehende Arbeitsdatei auf der Platte erstellt.

Arbeitsweise

In diesem Abschnitt wird die genaue *Arbeitsweise* des Programmes erläutert. Eine Übersicht zeigt das Bild 14.1. Zum besseren Verständnis ist es unbedingt erforderlich, die nachfolgend erklärten Begriffe zu kennen.

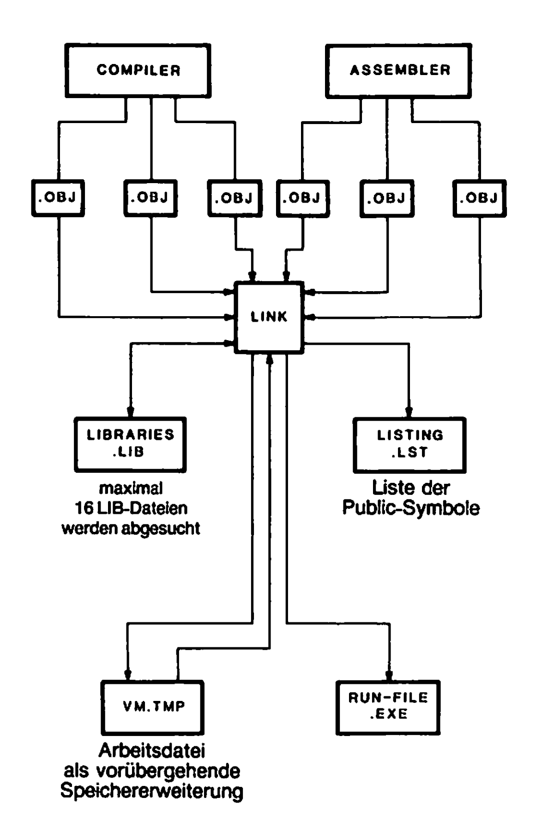


Bild 14.1
Arbeitsweise des Programmes LINK

Das Programm LINK erzeugt ein *ausführbares Programm,* indem es Befehle und Daten eines Programmes in den Arbeitsspeicher lädt. Die Reihenfolge wird dabei durch Bestimmungen in der Ursprungsdatei festgelegt. Sind alle Module eines neuen Programmes im Arbeitsspeicher richtig verbunden worden, kopiert LINK den Speicherinhalt in eine Datei. Zusätzlich werden in einem Programmkopf Anweisungen für das Betriebssystem gespeichert. Damit ist das Programm fertig.

Mit Hilfe besonderer Befehle kann LINK mitgeteilt werden, in welcher Reihenfolge alle Module zu laden sind. Einige Zusatzeingaben der Befehlszeile überschreiben diese Befehle wieder, falls gewünscht. Nachfolgend einige Erklärungen zu den Befehlen.

Definitionen

1. Segment Ein *Segment* ist ein zusammenhängender Bereich des Arbeitsspeichers bis zu einer Größe von 64 KB. Ein Segment kann überall im Speicher an einem Paragraphen (16-Byte-Grenze) beginnen. Der Inhalt eines Segmentes wird über ein Segmentregister plus einem Relativwert adressiert.

2. Gruppe Eine *Gruppe* ist eine Sammlung von überlappenden Segmenten, die in einem Bereich von 64 KB des Speichers vorhanden sind. Die Segmente der Gruppe werden durch den Assembler, den Compiler oder durch Sie benannt. Bei den höheren Programmiersprachen (BASIC, FORTRAN, COBOL, Pascal) wird dies durch den Compiler vorgenommen. Die Gruppe wird zum Adressieren von Segmenten im Speicher benutzt. Die Adresse jeder Gruppe wird durch ein Segmentregister bestimmt. Alle Segmente einer Gruppe haben die gleiche Startadresse. Adressen innerhalb der Gruppe werden durch einen Segmentregisterwert plus einem Relativwert bestimmt. Das Programm prüft nicht, ob die Beschränkung auf 64 KB eingehalten wird. Segmente einer Gruppe müssen nicht zusammenhängend in den Speicher geladen werden. Sie brauchen auch nicht derselben Klasse angehören. Alle Relativwerte innerhalb der Gruppe beziehen sich auf eine Segmentgrenze. Die Angabe eines Gruppennamens für ein Segment bestimmt nicht die Reihenfolge, in der das Segment geladen wird.

Arbeitsweise

3. Klasse

Eine *Klasse* ist eine Sammlung von Segmenten. Der Klassenname des Segmentes kontrolliert die Folge und relative Plazierung der Segmente im Speicher. Der Name der Klasse wird durch den Assembler, den Compiler oder durch Sie vergeben. Bei den höheren Programmiersprachen (BASIC, FORTRAN, COBOL, Pascal) wird dies durch den Compiler vorgenommen. Die Segmente der Klasse werden nacheinander in den Speicher geladen. Dies geschieht in der Reihenfolge, wie sie von LINK in den .OBJ-Dateien gefunden werden. Eine neue Klasse kann nur vor einer alten im Speicher plaziert werden, wenn ein Segment der ersten Klasse vor allen Segmenten der nächsten Klasse plaziert wird. Ein Beispiel:

```
DATAX SEGMENT 'DATA'
DATAX ENDS

TEXT  SEGMENT 'CODE
TEXT  ENDS

DATAZ SEGMENT 'DATA
DATAZ ENDS
```

Die beiden Segmente DATAX und DATAZ bilden eine Klasse, das Segment TEXT bildet eine andere Klasse. Alle Daten der Klasse DATA werden vor den Daten der Klasse CODE in den Speicher geladen. LINK kann eine unbegrenzte Zahl von Segmenten zu einer Klasse verbinden. Es besteht keine Beschränkung auf 64 KB.

4. Abgrenzung

Abgrenzungen beziehen sich auf bestimmte Grenzen. Dieses können Byte-, Word- oder Paragraphen- oder Seitengrenzen sein:

Byteabgrenzung Ein Segment kann an jeder Bytegrenze beginnen.

Wortabgrenzung Die Anfangsadresse eines Segmentes muß an einer geraden Adresse beginnen.

Paragraphen-abgrenzung Die Anfangsadresse eines Segmentes muß an einer 16-Byte-Grenze beginnen.

Seiten-abgrenzung Die Anfangsadresse eines Segmentes muß an einer 1024-Byte-Grenze beginnen.

5. Verbindungstyp Ein Verbindungstyp ist ein *Merkmal* eines Segmentes. Es zeigt dem Programm LINK, wie die Segmente eines Namens verbunden werden oder welche Eigenschaften das Segment besitzt. Verbindungstypen sind:

Public, Private, Memory, Stack und *Common.*

Die Arbeitsweise der Verbindungstypen wird im nächsten Abschnitt erklärt.

Segmentverbindung

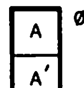
Das Programm LINK arbeitet mit vier *Verbindungstypen*, die im Ursprungsmodul vom Assembler oder Compiler definiert werden. (Der Speicherverbindungstyp Memory im Programm MACRO-86 wird genauso behandelt wie der Typ Public. LINK lädt nicht automatisch diesen Speicherverbindungstyp als höchstes Segment.)

Mit Hilfe der Verbindungstypen kombiniert LINK die Segmente folgendermaßen:

PRIVATE

Segmente mit dem Typ PRIVATE werden getrennt geladen und im Speicher abgelegt. Sie können zwar physikalisch hintereinander gespeichert werden, aber nicht logisch, auch nicht, wenn sie den gleichen Namen haben. Jedes Segment hat seine eigene Adresse. Ohne Eingabe eines Namens erhalten alle Segmente den Typ PRIVATE.





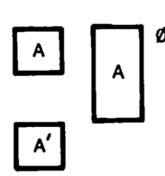


Weitere Dienstprogramme

Arbeitsweise

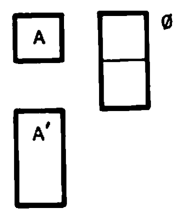
PUBLIC

Haben Segmente mit dem Typ PUBLIC den gleichen Namen und die gleiche Klasse, werden sie zusammenhängend geladen. Der Relativwert beginnt im ersten Segment und gilt bis zum letzten. Für alle Segmente des Typs PUBLIC, die den gleichen Namen und die gleiche Klasse haben, gibt es nur eine Basissegmentadresse. (Die Verbindungstypen STACK und MEMORY werden wie der Typ PUBLIC behandelt. Jedoch wird der Stapelzeiger auf die letzte Adresse des letzten Segmentes mit dem Typ STACK gesetzt.)



COMMON

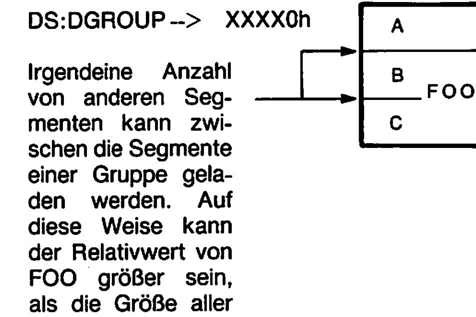
Haben Segmente mit dem Typ COMMON den gleichen Namen und die gleiche Klasse, werden Sie überlappend geladen. Es gibt nur eine Basisadresse für alle Segmente. Die Länge des Bereiches für den Typ COMMON wird durch das größte Segment definiert.



Plazieren Sie verschiedene Segmente in einer Gruppe, so errechnet der Assembler nur Relativadressen von einer Basisadresse für alle Segmente der Gruppe.

DS:DGROUP --> XXXX0h

Irgendeine Anzahl von anderen Segmenten kann zwischen die Segmente einer Gruppe geladen werden. Auf diese Weise kann der Relativwert von FOO größer sein, als die Größe aller Segmente von DGROUP zusammen, aber nicht größer als 64 KB.



0 – RELATIVWERT

Ein Operand des Relativwertes DGROUP:FOO enthält den Relativwert von FOO, bezogen auf das erste Segment von DGROUP (hier das Segment A).

Arbeitsweise

Segmente werden durch Klassennamen zusammengefaßt. Das Programm LINK lädt alle Segmente der ersten Klasse, die durch den Namen zusammengefaßt werden. Erst danach werden alle Segmente der nächsten Klasse komplett geladen. Ein Beispiel:

Das Programm enthält:	Geladen wird in dieser Reihenfolge:
A SEGMENT 'FOO'	'FOO'
B SEGMENT 'BAZ'	A
C SEGMENT 'BAZ'	E
D SEGMENT 'ZOO'	'BAZ'
E SEGMENT 'FOO'	B
	C
	'ZOO'
	D

Schreiben Sie Programme in Maschinensprache, können Sie bestimmen, in welcher Reihenfolge die Klassen geladen werden. Dies geschieht mit einer Musterdatei, in der Sie alle Klassen in der zu ladenden Reihenfolge eingeben. Geben Sie den Namen der Musterdatei als erste OBJ-Datei in der Befehlszeile an.

Diese Arbeitsweise darf nicht für BASIC, COBOL, FORTRAN oder PASCAL angewandt werden. Es ist die Aufgabe der Compiler.

Ein Beispiel für eine Musterdatei:

```
A    SEGMENT    'CODE'
A    ENDS

B    SEGMENT    'CONST'
B    ENDS

C    SEGMENT    'DATA'
C    ENDS

D    SEGMENT    STACK    'STACK'
D    ENDS

E    SEGMENT    'MEMORY'
E    ENDS
```

Arbeitsweise

Achten Sie darauf, alle Klassen Ihres Programmes zu benennen, sonst verlieren Sie die Kontrolle über den Ladevorgang der Klassen. (Die Segmente werden, nach Klassennamen geordnet, in der Reihenfolge ihres Auffindens geladen.)

Bestimmung der Speicheradressen

Der Mikroprozessor 8088 kann maximal 1.048.575 Bytes (1 MB) adressieren. Zur Darstellung dieser Zahl benötigt man *20 Bit*. Da der Prozessor jedoch nur *16-Bit-Worte* verarbeiten kann, kann er damit nur 65 535 Bytes (64KB) sofort bearbeiten. Um den vollen Speicherraum adressieren zu können, wird die Teiladressierung benutzt.

Die Adresse besteht jetzt aus einer *Basisadresse* (Segmentadresse) und einer *relativen Adresse* zur Basisadresse (Offset). Beide Zahlen werden in hexadezimaler Schreibweise angegeben. Werden beide Adressen addiert, erhält man die absolute Adresse. Dabei wird der Basisadresse vor dem Rechenvorgang eine Null angefügt (vier binäre Nullen), danach wird die relative Adresse dazu addiert. Ein Beispiel:

Zur Segmentadresse F0C2h soll die relative Adresse 01B5h addiert werden:

```
F0C20h          (Segment um vier Bit verschoben)
 01B5h          (Relativwert)
------
F0DD5h          (absolute Adresse)
```

LINK bestimmt *Segmentadressen*, indem es jedem Segment einen Segment- und einen Klassennamen zuteilt. Das Programm bestimmt eine Rahmenadresse (beginnend bei 0:0) und darauf aufbauend eine relative Adresse. Mit dieser Information werden verschiebbare Referenzen gelöst.

Verschiebbare Bestimmungen

LINK errechnet für folgende Referenzen die neuen Adressen:

Kurzreferenzen

Alle Kurzreferenzen sind *relativ*. Die Rahmenadressen des Ursprungs- und Zielsegmentes müssen gleich sein. Sind sie verschieden oder ist das Ziel mehr als 128 Bytes vom Ursprung entfernt, wird die Fehlermeldung

Fixup offset exceeds field width	Relativbestimmung ist zu groß

ausgegeben. Das Ergebnis der Kurzreferenz darf nicht größer sein, als der maximal darstellbare Wert eines Byte mit Vorzeichen.

Relative Nahreferenzen

Für Nahreferenzen sind die *Rahmenadressen* des Ursprungs- und Zielsegmentes gleich. Sind sie verschieden oder ist das Ziel mehr als 32 KB vom Ursprung entfernt, wird die Fehlermeldung

Fixup offset exceeds field width	Relativbestimmung ist zu groß

ausgegeben. Das Ergebnis der Nahreferenz darf nicht größer sein, als der maximal darstellbare Wert eines Wortes mit Vorzeichen (16 Bit).

Absolute Nahreferenzen

LINK löst *absolute Nahreferenzen,* wenn Daten innerhalb eines Segmentes oder einer Gruppe erreicht werden müssen. Dafür wird ein Relativwert errechnet, der nicht größer als 64 KBytes sein darf, sonst wird die Fehlermeldung

Fixup offset exceeds field width	Relativbestimmung ist zu groß

ausgegeben. Das Ergebnis der absoluten Nahreferenz darf nicht größer sein, als der maximal darstellbare Wert eines Wortes (16 Bit).

Fernreferenzen

Bei Fernreferenzen ist die *Zieladresse* in einem anderen Rahmen. Die Rahmenadressen dürfen jedoch nicht mehr als 64 KB auseinander sein. Ist der relative Wert im definierten Rahmen größer als 64 KB oder die Adresse des Zielsegmentes nicht durch das Rahmensegment adressierbar, wird die Fehlermeldung

Fixup offset exceeds field width	Relativbestimmung ist zu groß

ausgegeben. Das Ergebnis der Fernreferenz muß aus einer Segmentadresse und einer Relativadresse bestehen.

Dateien von LINK

Das Programm arbeitet mit ein oder zwei *Eingabedateien* und erstellt zwei *Ausgabedateien.* Falls erforderlich, wird eine vorübergehende Arbeitsdatei erstellt. Das Programm kann maximal sechzehn LIB-Dateien absuchen. Jede Datei wird durch ihren Namenszusatz identifiziert. Geben Sie keinen Namenszusatz an, wird vom Programm der in Tabelle 14.1 gezeigte Zusatz angenommen. Jedem Namen darf eine Laufwerkbezeichnung und/oder ein Verzeichnisname vorangestellt werden.

Tabelle 14.1 Dateien von LINK

ART	DATEI	ZUSATZ
Ein	Objekt	.OBJ
Ein	Bibliothek	.LIB
Aus	Run	.EXE (darf nicht verändert werden)
Aus	Liste	.MAP (darf verändert werden)
Ein/Aus	Arbeitsdatei	.TMP (kann nicht geändert werden)

Arbeitsdatei VM.TMP

Das Programm LINK benutzt während des Ablaufes den gesamten Arbeitsspeicher. Wird durch die zu verbindenden Dateien eine größere Ausgabedatei erstellt, als Speicherkapazität vorhanden ist, erzeugt das Programm eine Arbeitsdatei. Diese Datei erhält den Namen *VM.TMP*. Sie dient vorübergehend als *Speichererweiterung*. In ihr werden Informationen über das neue Modul abgespeichert, bevor diese in die neue Datei geschrieben werden. Dies wird durch die Meldung

VM.TMP has been created. Do not change diskette in drive X:	Datei VM.TMP wurde angelegt. Diskette in Laufwerk X: nicht wechseln.

angezeigt. Erscheint diese Meldung, dürfen Sie die Diskette bis zum Ende des Programmes nicht mehr wechseln. Andernfalls wird das Programm sofort abgebrochen. Es erscheint dann folgende Meldung:

Unexpected end of file on VM.TMP	Datei VM.TMP nicht mehr vorhanden

LINK löscht die Datei am Ende des Programmablaufes wieder. Sollten Sie eine Ihrer Dateien mit diesem Namen benannt haben, überschreibt LINK diese Datei und löscht sie anschließend ebenfalls.

Programmaufruf

Syntax

[d:] [Weg] Link
[d:] [Weg] Link OBJ-Datei, EXE-Datei, MAP-Datei, LIB-Datei [/X]
[d:] [Weg] Link @Dateibez

dabei bedeutet:

d:	vor dem Programmnamen bestimmt das aufzurufende Laufwerk
Weg	vor dem Programmnamen bestimmt das aufzurufende Verzeichnis
OBJ-Datei	Namen aller zu verbindenden Objektmodule. Diese werden durch ein Pluszeichen (+) miteinander verbunden.
EXE-Datei	Name der neuen ausführbaren Datei.
MAP-Datei	Name der Datei für gelöste Querreferenzen und Fehler.
LIB-Datei	Namen aller abzusuchenden Bibliotheken (maximal sechzehn).
/X	Kombination der folgenden Zusatzeingaben:

/CPARMAXALLOC:n	Bestimmt die Größe des benötigten Arbeitsspeichers für das Programm in Paragraphen (16 Bytes). n ist eine ganze Zahl im Bereich 1 bis 65 535.
/DOSSEG	Ordnet die Segmente in der Form: Code, Data.

/DSALLOCATE	Es werden alle Daten an das hohe Ende des Datensegmentes geladen.
/HIGH	Die Ausführungsdatei wird so hoch wie möglich in den Speicher geladen. Dieser Zusatz kann mit dem Zusatz /D zusammen eingegeben werden.
/LINENUMBERS	In der MAP-Datei werden Zeilennummern und Adressen der Quelldatei angezeigt.
/MAP	In der MAP-Datei werden alle im Eingabemodul als extern definierte Symbole mit Adressen alphabetisch aufgelistet.
/NODEFAULTLIBRARY	Es werden keine vorgegebenen LIB-Dateien abgesucht.
/NOGROUPASSOCIATION	Es werden keine Gruppenanweisungen der Objektmodule beachtet.
/NOIGNORECASE	Symbole mit Groß- und Kleinbuchstaben werden verschieden behandelt.
/OVERLAYINTERRUPT:n	Bestimmt den Interrupt zum Nachladen von Overlaydateien. n ist eine ganze Zahl im Bereich 1 bis 255.
/PAUSE	Das Programm wird angehalten und wartet auf die Eingabe von RETURN.
/SEGMENTS:n	Bestimmt die maximale Zahl der zu bearbeitenden Segmente. n ist eine ganze Zahl im Bereich 1 bis 1024.
/STACK:n	Mit der Eingabe wird die Größe des Stapelsegmentes bestimmt. n ist eine ganze Zahl im Bereich 1 bis 255.

Bemerkungen

Während des *Programmablaufes* werden von Ihnen zwei verschiedene Eingaben gefordert:

- Programmaufruf
- Antwort auf eine Programmanfrage

Durch dreizehn *Zusatzeingaben* haben Sie die Möglichkeit, bestimmte Abläufe des Programmes zu beeinflussen. Sie können dabei wählen, ob Sie alle Eingaben über die Tastatur vornehmen, oder ob Sie alle Eingaben in einer separaten Datei speichern und die Datei vom Programm abfragen lassen.

Programmaufruf

Das Programm kann von Ihnen auf *drei* verschiedene Arten aufgerufen werden. Sie antworten der Reihe nach auf alle vom Programm gestellten Anforderungen. Sie schreiben alle Antworten in die Befehlszeile mit der das Programm gestartet wird. Sie schreiben alle Antworten in eine Datei, die Sie vorher erstellen.

Tabelle 14.2 Übersicht LINK-Programmaufruf

ART	EINGABE
Einzelne Antwort	LINK
Antwort in der Befehlszeile	LINK Dateiname [/X]
Antwortdatei	LINK @Dateiname

Einzelne Antwort

Rufen Sie das Programm nur mit seinem Namen auf, geben Sie in der Befehlszeile ein:

```
LINK
```

Das Programm wird in den Speicher geladen. Die zur Bearbeitung benötigten Dateien werden jetzt einzeln durch das Programm abge-

fragt. Wollen Sie die vorgegebenen Werte übernehmen, beantworten Sie die Fragen mit **RETURN**. Zu jedem Dateinamen kann ein Zusatz eingegeben werden. Die in den Tabellen 14.3 und 14.4 gezeigten Möglichkeiten werden im späteren Abschnitt dieses Kapitels erklärt.

Tabelle 14.3 LINK-Antworten

FRAGE	ANTWORT
Object Modules [.OBJ]:	Eingabe der zu verbindenden Module (Name, Laufwerk, Bezeichnung). Die Module können durch Leertaste oder Pluszeichen verbunden werden. Wird das Pluszeichen zuletzt in der Zeile eingegeben, erscheint die Anfrage erneut. Eine Eingabe wird immer verlangt.
Run File [Objfile.EXE]:	Eingabe des Namens für das neue Modul. Ohne Eingabe wird der erste Name der Objektdateien gewählt und mit dem Zusatz .EXE versehen.
List File [NUL.MAP]:	Eingabe des Dateinamens für die Liste. Die Liste wird nur erstellt, wenn hier ein Name eingegeben wird.
Libraries [.LIB]:	Eingabe der zu durchsuchenden Bibliotheken. Die Dateinamen werden durch Leertaste oder Pluszeichen verbunden. Wird das Pluszeichen zuletzt in der Zeile eingegeben, erscheint die Anfrage erneut.

Programmaufruf

Tabelle 14.3 LINK-Zusatzeingaben

EINGABE	FUNKTION
/CPARMAXALLOC:n /C	Der Wert dieser Zusatzeingabe wird an das Betriebssystem weitergereicht. DOS reserviert für die Datei während des Ladevorganges einen entsprechenden Bereich des Arbeitsspeichers. Mit n wird die Zahl der Paragraphen für das Programm bestimmt.
/DOSSEG /DO	Mit diesem Zusatz weisen Sie LINK an, alle Segmente der Module in der Reihenfolge CODE, DATA zu laden. Sollten andere Anweisungen vorhanden sein, werden diese nicht beachtet. LINK richtet sich hierbei immer nach dem Klassennamen der Segmente.
/DSALLOCATE /DS	Es werden alle Daten an das hohe Ende des Datensegmentes geladen. Dieser Zusatz ist für PASCAL- und FORTRAN-Programme erforderlich.
/HIGH /HI	Die Ausführungsdatei wird so hoch wie möglich in den Speicher geladen. Dieser Zusatz kann mit dem Zusatz /DS zusammen eingegeben werden. Er darf nicht für PASCAL- und FORTRAN-Programme benutzt werden.
/LINENUMBERS /LI	In der .MAP-Datei werden Zeilennummern und Adressen der Quelldatei angezeigt.
/MAP /M	In der .MAP-Datei werden alle im Eingabemodul als extern definierten Symbole mit Adressen alphabetisch aufgelistet.
/NODEFAULTLIBRARY /NOD	Geben Sie diesen Zusatz ein, beachtet LINK nicht die LIB-Dateien, die im Objektmodul gespeichert sind. Damit können Sie die Angaben eines Compilers überschreiben und Ihre eigenen Bibliotheken benennen.

Tabelle 14.3 LINK-Zusatzeingaben

EINGABE	FUNKTION
/NOGROUPASSOCIATION /NOG	Nach Eingabe dieses Zusatzes ignoriert LINK alle Gruppenbestimmungen bei der Errechnung von Speicheradressen für Daten- und Programmvariable.
/NOIGNORECASE /NOI	Mit diesem Zusatz weisen Sie LINK an, Variable mit gleichem Namen aber unterschiedlicher Groß- Kleinschreibung als verschieden zu behandeln.
/OVERLAYINTERRUPT:n /O	Dieser Zusatz bestimmt den Software-Interrupt, mit dem Overlaymodule aufgerufen werden. Ohne Eingabe ist dies Interrupt 3Fh. n ist eine ganze Zahl im Bereich 1 bis 255.
/PAUSE /P	Bevor die neue EXE-Datei fertig ist, wird das Programm angehalten. Sie können dann die Diskette wechseln. Nach Eingabe von RETURN schreibt LINK die Datei auf Platte.
/SEGMENTS:n /SE	Dieser Zusatz bestimmt die maximale Zahl der zu bearbeitenden Segmente für die neue Datei. n ist eine ganze Zahl im Bereich 1 bis 1024. Ohne Eingabe beträgt der Wert 128.
/STACK:n /ST	Mit der Eingabe wird die Größe des Stapelsegmentes bestimmt.

Programmaufruf

Zeichen der Befehlszeile

In der Befehlszeile dürfen drei Zeichen verwendet werden:

\+ Das Pluszeichen trennt die Namen der Objektmodule (ebenso wie die Leertaste). Geben Sie als letztes Zeichen der Zeile das Pluszeichen ein, wird die logische Zeile in der nächsten physikalischen Zeile fortgesetzt. Dies ist erforderlich bei mehreren langen Objektmodulnamen. Das Programm fordert Sie dann auf, in der nächsten Zeile weitere Modulnamen einzugeben. Achten Sie darauf, nach dem letzten Modulnamen keine weiteren Zeichen einzugeben. Ein Beispiel:

```
Object Modules [.OBJ]: FUN TEXT TABLE CARE+
Object Modules [.OBJ]: FOO+FLIP+FLOP+
Object Modules [.OBJ]: CORSAIR
```

; Ab der Namensanfrage für die ausführende Datei (Run File) bestimmen Sie mit dem Semikolon als Antwort, daß die jeweils vorgegebenen Werte übernommen werden. Dies gilt für alle weiteren Anfragen. Die Eingabe der Dateinamen ist damit beendet. Ein Beispiel:

```
Object Modules [.OBJ]: FUN TEXT TABLE CARE
Run Module [FUN.EXE]:;
```

CTRL-BREAK Mit der Tastenkombination **CTRL-BREAK** kann das Programm jederzeit abgebrochen werden. Geben Sie falsche Dateinamen ein, können Sie diese Namen in der Arbeitszeile mit den Funktionstasten korrigieren. Andere Namen können nicht mehr geändert werden. Brechen Sie das Programm ab, und starten Sie neu. Auf dem Bildschirm erscheint die Meldung:

```
Status 1
```

Das Programm übergibt damit gleichzeitig einen *Fehlerkode* an das Betriebssystem. Der Wert ist 1 und kann in einer Batchdatei mit dem Befehl ERRORLEVEL abgefragt werden.

Antwort in der Befehlszeile

Rufen Sie das Programm gleichzeitig mit allen zu bearbeitenden Namen auf, geben Sie folgende Befehlszeile ein:

LINK Liste der .OBJ-Dateien,.EXE-Datei,.MAP-Datei,Liste der .LIB- Dateien /X...

dabei bedeutet:

Liste der .OBJ-Dateien	Namen aller zu verbindenden Objektmodule. Diese werden durch ein Pluszeichen miteinander verbunden.
.EXE-Datei	Name der neuen ausführbaren Datei
.MAP-Datei	Name der Datei für extern definierte Symbole und Fehler
Liste der .LIB-Dateien	Namen aller abzusuchenden Bibliotheken (maximal sechzehn)
/X	Zusatzeingaben (s. a. Tabelle 14.3)

Sämtliche *Eingaben* nach dem Programmnamen LINK werden als Antworten auf die Anfragen gewertet. Die einzelnen Eingabefelder müssen durch ein Komma getrennt werden. Zu jedem Dateinamen kann eine Laufwerkbezeichnung und/oder der Name eines Verzeichnisses mit eingegeben werden. Soll für ein Feld der vorgegebene Wert übernommen werden, geben Sie ein weiteres Komma ein. Ein Beispiel:

LINK FUN+TEXT+TABLE+CARE/P/M,,FUNLIST,COBLIB.LIB

In diesem Beispiel werden nach dem Programmstart die Module FUN.OBJ, TEXT.OBJ, TABLE.OBJ und CARE.OBJ geladen. LINK verbindet dann alle Module. Danach wird das Programm angehalten, und Sie können eine neue Diskette in das Laufwerk legen. Erst nach Betätigen der Taste **RETURN** wird LINK fortgesetzt. Die Ausführungsdatei bekommt den Namen FUN.EXE. Alle globalen Symbolnamen werden in die Datei FUNLIST geschrieben. Die Bibliothek COBLIB.LIB wird abgesucht.

Antwortdatei

Ihre Antworten auf die Befehlsanfragen von LINK können in einer separaten *Textdatei* gespeichert werden. Das erste Zeichen der Datei muß immer „@" sein. Nur daran erkennt LINK, daß es die Antwortdatei ist. Rufen Sie das Programm folgendermaßen auf:

```
LINK @Dateibez
```

Dem Namen der *Antwortdatei* kann ein Laufwerk und/oder der Name eines Verzeichnisses vorangestellt werden. Sie können diese Datei mit dem Programm EDLIN erstellen oder auch direkt über die Tastatur eingeben.

Die Antwortdatei muß alle Antworten in der korrekten Reihenfolge enthalten. Jede Antwort muß in einer eigenen Zeile stehen. Längere Antworten können über mehrere Zeilen eingegeben werden, wenn Sie diese Zeilen durch das Pluszeichen verbinden.

Nach dem *Programmaufruf* werden sämtliche Fragen und Antworten zur Kontrolle auf dem Bildschirm angezeigt. Enthält die Datei auf eine Frage keine Antwort, wird die Abarbeitung der Datei unterbrochen. Sie haben die Möglichkeit, eine Antwort einzugeben. Danach wird die Abarbeitung fortgesetzt. Ein Beispiel:

Die Antwortdatei enthält folgende Zeilen:

FUN TEXT TABLE CARE
/PAUSE/MAP
FUNLIST
COBLIB.LIB

In diesem Beispiel werden nach dem Programmstart die Module FUN.OBJ, TEXT.OBJ, TABLE.OBJ und CARE.OBJ geladen. LINK verbindet dann alle Module. Danach wird das Programm angehalten, und Sie können eine neue Diskette in das Laufwerk legen. Erst nach Betätigen der Taste **RETURN** wird LINK fortgesetzt. Die Ausführungsdatei bekommt den Namen FUN.EXE. Alle globalen Symbolnamen werden in die Datei FUNLIST geschrieben. Die Bibliothek COBLIB.LIB wird abgesucht.

Programmablauf

Nach dem Starten des Programmes LINK werden Sie aufgefordert, bis zu vier verschiedene Dateinamen einzugeben. Die *Abfrage* erfolgt schrittweise. Nach der letzten Eingabe wird mit der Programmausführung begonnen. Tritt kein Fehler auf, wird das Programm normal beendet, und Sie sind wieder im Betriebssystem. Findet LINK einen Fehler, erscheint die entsprechende Fehlermeldung und das Programm wird abgebrochen. Nach Behebung des Fehlers müssen Sie LINK neu starten.

Befehlsanfragen

Bei der Anfrage der Dateinamen werden Ihnen die vorgegebenen Werte in eckigen Klammern ([ ]) angezeigt. Durch Betätigen der Taste **RETURN** übernehmen Sie diese Werte.

Objektmodule

[.OBJ]:

Geben Sie hier die Namen der zu verbindenden Module ein. Fehlt der Namenszusatz, wird vom Programm OBJ eingesetzt. Die Module werden durch Pluszeichen miteinander verbunden. Geben Sie die Modulnamen in der Reihenfolge ein, in der Klassen und Segmente geladen werden sollen. Verbinden Sie Module eines Compilers mit Modulen anderer Programmiersprachen, geben Sie die Compilermodule immer zuerst ein.

Ausführungsdatei

[Objfile.EXE]:

Geben Sie hier den Namen der neuen Ausführungsdatei an. Der Zusatz ist immer .EXE (andere Eingaben werden ignoriert). Geben Sie keine Antwort ein, wird der Name des ersten Objektmodules übernommen. Ein Beispiel:

Run File [FUN.EXE]: B:ROLL/P

Programmablauf

Die neue Datei wird ROLL.EXE genannt und auf Laufwerk B: geschrieben. Vorher wird das Programm angehalten. Sie haben die Möglichkeit, die Diskette zu wechseln.

Listdatei

[NUL.MAP]:

Die Listdatei enthält für jedes Segment des Objektmodules einen Eintrag. Jeder Eintrag zeigt ebenfalls den relativen Speicherplatz in der Ausführungsdatei. Vorgegeben ist keine Datei (NUL). LINK erstellt auch dann eine Datei, wenn hier der Zusatz /MAP eingegeben wird. Geben Sie nur einen Namen ein, könnte die Datei folgenden Inhalt haben:

```
Start   Stop    Länge  Name          Klasse
00000H  00D2DH  0D2EH  BIOS          BIOS
00D30H  01523H  07F4H  SYSINITSEG    SYSTEM_NIT
```

Bibliotheken

[.LIB]:

Geben Sie hier bis zu sechzehn Namen von LIB-Dateien ein. (**RETURN** bedeutet keine Suche nach Bibliotheken.) Die Dateien müssen vom Dienstprogramm LIB erstellt worden sein. Die einzelnen Namen werden durch ein Pluszeichen miteinander verbunden. LINK durchsucht die Dateien in der angegebenen Reihenfolge, um externe Referenzen zu lösen. Findet es das Symbol in einer Datei, so formt LINK daraus ein weiteres Objektmodul. Ist eine Datei nicht vorhanden, erscheint die Meldung:

```
Cannot find library ....     Datei nicht gefunden
Enter new drive letter:      Geben Sie ein neues Laufwerk:
```

Geben Sie dann nur den Buchstaben des entsprechenden Laufwerkes oder einen Verzeichnisnamen ein. Sie können auch mit dem MS-DOS-Befehl SET ein oder mehrere Verzeichnisse bestimmen, die dann von LINK abgesucht werden. Geben Sie z. B. ein:

```
SET LIB=C:\SYSTEM\LIB
```

Jetzt sucht LINK immer im Verzeichnis \SYSTEM\LIB des Laufwerkes C: nach den LIB-Dateien.

Die Dateien werden von LINK nicht sequentiell abgesucht. Es wird über einen vorangestellten Index auf das Symbol zugegriffen. Dadurch wird die Bearbeitungszeit erheblich verkürzt.

Zusatzeingaben

Es ist möglich, über dreizehn Zusatzeingaben die Arbeitsweise von LINK zu beeinflussen. Die Eingaben werden immer als letzter Teil einer Antwort eingegeben. Mehrere Eingaben zugleich sind erlaubt. Jeder Zusatz wird durch einen vorangestellten Schrägstrich (/) gekennzeichnet. Die Zusätze dürfen mit einem oder mehreren Buchstaben angekürzt werden. Ein Beispiel:

RICHTIG	FALSCH
/D	/DSL
/DS	/DAL
/DSA	/DLC
/DSALLOCA	/DSALLOCT

/CPARMAXALLOC:n

Mindesteingabe: /C

Der Wert dieser Zusatzeingabe wird an das *Betriebssystem* weitergereicht. MS-DOS reserviert normalerweise den gesamten freien Arbeitsspeicher für die Datei während des Ladevorganges. Damit können anschließend keine weiteren Programme mehr geladen werden. Mit *n* bestimmen Sie die Zahl der Paragraphen, die das Programm unbedingt benötigt. Der darüberliegende Speicher bleibt frei. Geben Sie hier einen zu kleinen Wert ein, wird er ignoriert. Der Wert für /C muß mindestens der Größe des Programmes entsprechen.

Programmablauf

/DOSSEG

Mindesteingabe: /DO

Mit diesem Zusatz weisen Sie LINK an, alle *Segmente* der Module in der Reihenfolge CODE, DATA zu laden. Sollten andere Anweisungen vorhanden sein, werden diese nicht beachtet. LINK richtet sich hierbei immer nach dem Klassennamen der Segmente in der Reihenfolge:

1. Alle Segmente mit dem Klassennamen 'CODE' werden zuerst geladen.

2. Alle Segmente mit einem anderen Klassennamen als 'DGROUP' werden anschließend geladen.

3. Alle Segmente mit dem Klassennamen 'DGROUP' werden zuletzt geladen.

/DSALLOCATE

Mindesteingabe: /DS

Dieser Zusatz veranlaßt LINK, alle *Daten* (der Gruppe DGROUP) an das hohe Ende des Datensegmentes zu laden (ab der Adresse 0FFFFh). Ohne Eingabe werden die Daten an das niedrige Ende geladen (Adresse 0000h). Beim Ablauf wird der DS-Zeiger auf die niedrigste freie Adresse gesetzt. Das gesamte DS-Segment kann dann genutzt werden. Geben Sie den Zusatz /DSALLOCATE ohne den Zusatz /HIGH ein, erlauben Sie Ihrem Programm, den freien Speicherplatz, der unterhalb des bestimmten DS-Segmentes liegt, dynamisch zu nutzen. Der Platz kann immer noch durch den DS-Zeiger erreicht werden. Dieser Zusatz wird für PASCAL- und FORTRAN-Programme benötigt.

/HIGH

Mindesteingabe: /HI

Dieser Zusatz veranlaßt LINK in der Ausführungsdatei zusätzliche Informationen zu speichern, damit DOS das Programm so hoch wie möglich in den Arbeitsspeicher lädt. Ohne diesen Zusatz wird die Datei immer so tief wie möglich geladen. Der Zusatz darf zusammen mit dem Zusatz /DSALLOCATE eingegeben werden. Er darf nicht für PASCAL- und FORTRAN-Programme benutzt werden.

/LINENUMBERS

Mindesteingabe: /LI

Dieser Zusatz veranlaßt LINK, in der Datei *.MAP* Zeilennummern und Adressen der Textdatei mit auszudrucken. Das ist nur möglich bei Objektmodulen, die von Compilern mit diesen Informationen erstellt wurden. Bei allen anderen Modulen wird dieser Zusatz ignoriert.

/MAP

Mindesteingabe: /M

Dieser Zusatz veranlaßt LINK, in der Datei *.MAP* alle allgemeinen Symbole aufzulisten. Geben Sie den Zusatz nicht ein, werden nur Segmentdefinitionen und Fehler (nicht definierte Symbole) aufgelistet. Die Symbole werden alphabetisch mit ihrer Segment- und Relativadresse in der Ausführungsdatei aufgelistet. Diese Liste erscheint am Ende der .MAP-Datei. Ein Beispiel:

Start	Stop	Länge	Name	Klasse
00000H	00D2DH	0D2EH	BIOS	BIOS
00D30H	01523H	07F4H	SYSINITSEG	SYSTEM_INIT

Programmablauf

Adresse	Publicsymbole, alphabetisch
00D3:005B	BADCOM
00D3:0048	BADCOUNTRY
00D3:0000	BADOPM
00D3:0082	BUFFERS
00D3:0075	CURRENT_DOS_LOCATION
00D3:0081	DEFAULT_DRIVE
00D3:007B	DEVICE_LIST
00D3:0083	FILES
00D3:0079	FINAL_DOS_LOCATION
00D3:007F	MEMORY_SIZE
00D3:0070	SYSINIT
00D3:006F	SYSSIZE

Adresse	Publicsymbole, numerisch
00D3:0000	BADOPM
00D3:0048	BADCOUNTRY
00D3:005B	BADCOM
00D3:006F	SYSSIZE
00D3:0070	SYSINIT
00D3:0075	CURRENT_DOS_LOCATION
00D3:0079	FINAL_DOS_LOCATION
00D3:007B	DEVICE_LIST
00D3:007F	MEMORY_SIZE
00D3:0081	DEFAULT_DRIVE
00D3:0082	BUFFERS
00D3:0083	FILES

/NODEFAULTLIBRARYSEARCH

Mindesteingabe: /NOD

Dieser Zusatz veranlaßt LINK, nicht die vorgegebenen *Bibliotheken* in den Objektmodulen abzusuchen. Verschiedene Compiler speichern im Modul den Namen einer Bibliothek, damit LINK unbedingt danach sucht. Geben Sie den Zusatz /NOD ein, können Sie dies überschreiben und Ihre eigenen Bibliotheken absuchen lassen. Verbinden Sie z. B. Pascalmodule und geben den Zusatz /NOD ein, wird nicht in der Bibliothek PASCAL.LIB für die Lösung einer externen Querreferenz gesucht.

/NOGROUPASSOCIATION

Mindesteingabe: /NOD

Nach Eingabe dieses Zusatzes ignoriert LINK alle *Gruppenbestimmungen* bei der Errechnung von Speicheradressen für Daten- und Programmvariable. Diesen Zusatz sollten Sie nicht benutzen.

/NOIGNORECASE

Mindesteingabe: /NOI

Mit diesem Zusatz weisen Sie LINK an, *Variable* mit gleichem Namen aber unterschiedlicher Groß- und Kleinschreibung als verschieden zu behandeln. Normalerweise formt LINK alle Variablennamen in Versalien um. Verschiedene Compiler (z. B. C) dagegen behandeln Variable nur als gleich, wenn auch die Groß- und Kleinschreibung übereinstimmt. In diesem Falle muß auch LINK dies beachten.

/OVERLAYINTERRUPT:n

Mindesteingabe: /O:n

Dieser Zusatz bestimmt den *Software-Interrupt* mit dem Overlaymodule aufgerufen werden. Schließen Sie diese Module in der Befehlszeile in runde Klammern ein. Der Interrupt kann als Dezimalwert oder als Hexadezimalwert (mit vorangestelltem „0X") eingegeben werden. Ohne Eingabe ist dies Interrupt 3Fh. n ist eine ganze Zahl im Bereich 1 bis 255. Sie sollten jedoch niemals einen Wert kleiner als 3Fh benutzen, da diese Interrupts für MS-DOS reserviert sind. Ein Beispiel:

```
LINK TEST+(CLS) /O:0XFF;
```

Es werden die Module TEST und CLS verbunden. Der Software-Interrupt für die Overlaydatei CLS wird mit 255 (0FFh) bestimmt.

Programmablauf

/PAUSE

Mindesteingabe: /P

Dieser Zusatz veranlaßt LINK, die *Abarbeitung* zu stoppen, wenn die Ausführungsdatei auf die Diskette geschrieben werden soll. Damit haben Sie die Möglichkeit, die Diskette zu wechseln. Es erscheint folgende Meldung:

About to generate .EXE file Change disks <hit ENTER>	Die neue Datei kann geschrie- ben werden, wechseln Sie die Diskette <Betätigen Sie ENTER>

Wurde vorher die Datei VM.TMP angelegt, darf jetzt **NICHT** die Diskette gewechselt werden, sonst wird der Vorgang sofort abgebrochen.

/SEGMENTS:n

Mindesteingabe: /SE:n

Dieser Zusatz bestimmt die maximale Zahl der zu bearbeitenden *Segmente* für die neue Datei. *n* ist eine ganze Zahl im Bereich 1 bis 1024. Ohne Eingabe beträgt der Wert 128. Benutzen Sie diesen Zusatz nur, wenn die zu verbindenden Module mehr als 128 Segmente beinhalten.

/STACK:n

Mindesteingabe: /ST:n

Mit diesem Zusatz können Sie eine beliebige Zahl zwischen 200h und FFFFh eingeben. Damit bestimmen Sie die Größe des *Stapelsegmentes.* Ohne Eingabe wird dieser Wert von LINK automatisch errechnet, sofern wenigstens ein Objektmodul dazu die nötigen Informationen hat. Ist das nicht der Fall, erscheint die Meldung:

Warning: NO STACK SEGMENT	Warnung: KEIN STAPELSEGMENT

Das Programm wird nach dieser Meldung normal weiterbearbeitet. Soll das fertige Modul von der Datei EXE2BIN umgeformt werden, darf es kein Stacksegment enthalten. In diesem beachten Sie die Warnung nicht.

Fehlermeldungen

Das Programm wird durch jede Fehlermeldung (Ausnahme: No Stack segment) sofort abgebrochen. Es muß anschließend neu gestartet werden.

Attempt to access data outside of segment bounds, possibly bad object module	Es wird der Versuch gemacht, auf Daten außerhalb der Segmentadresse zuzugreifen, möglicherweise ist das Modul fehlerhaft

ERKLÄRUNG:
Das Objektmodul ist fehlerhaft. Korrigieren Sie den Fehler im Modul, und starten Sie LINK neu.

Bad value for cparMaxAlloc	Falscher Wert für CPARMAXALLOC

ERKLÄRUNG:
Sie haben für diesen Zusatz einen falschen numerischen Wert eingegeben. Erlaubt sind Zahlen im Bereich 1 bis 65 535.

Cannot find file	Die Datei ist nicht vorhanden

ERKLÄRUNG:
Der eingegebene Dateiname ist nicht auf der Diskette. Geben Sie den Namen richtig ein, oder tauschen Sie die Diskette.

Cannot nest response file	Eingabedatei darf nicht verschachtelt werden

ERKLÄRUNG:
Die Datei mit den Eingaben für LINK ruft eine weitere Eingabedatei auf. Dies ist nicht erlaubt.

Fehlermeldungen

Cannot open response file	Eingabedatei kann nicht gelesen werden

ERKLÄRUNG:
Die Datei mit den Eingaben für LINK ist fehlerhaft und kann vom Programm nicht gelesen werden.

Cannot open temporary file	Die Arbeitsdatei kann nicht angelegt werden

ERKLÄRUNG:
Die Datei VM.TMP, die zur Speichererweiterung benötigt wird, kann nicht angelegt werden, weil die Diskette voll ist. Nehmen Sie eine andere Diskette.

Error: Dup record too complex	Dup-Satz zu groß

ERKLÄRUNG:
Der Dup-Satz im Modul ist zu groß. Ändern Sie das Modul, und starten Sie LINK neu.

Error: Fixup offset exceeds field width	Fehler: Relativbestimmung ist zu groß

ERKLÄRUNG:
Eine Anweisung der Maschinensprache zeigt auf eine Adresse außerhalb des Segmentes. Ändern Sie das Modul.

Incorrect DOS-Version, use 2.0 or later	Falsche DOS-Version

ERKLÄRUNG:
Das Programm benötigt das Betriebssystem MS-DOS 2.0 oder höher.

Input file read error	Lesefehler der Datei

ERKLÄRUNG:
Das Modul ist fehlerhaft, eventuell ein Fehler der Diskette. Überprüfen Sie das Modul.

Insufficient stack space	Kein Platz für das Stapelsegment vorhanden

ERKLÄRUNG:
Der Arbeitsspeicher ist nicht mehr groß genug für LINK. Sie benötigen mehr Speicher oder haben zu viele residente Programme als Erweiterung des Betriebssystemes geladen.

Interrupt number exceeds 255	Interrupt größer als 255

ERKLÄRUNG:
Sie haben für den Zusatz /OVERLAYINTERRUPT einen zu großen Wert eingegeben. Es sind nur Zahlen im Bereich 1 bis 255 erlaubt.

Invalid numeric switch specification	Falscher Zahlenwert

ERKLÄRUNG:
Der eingegebene Wert ist kein gültiger Hexadezimalwert. Starten Sie LINK erneut mit dem richtigen Wert.

Invalid object module	Fehlerhaftes Modul

ERKLÄRUNG:
Das Objektmodul ist nicht richtig erstellt oder unvollständig. Überprüfen Sie das Modul.

Nested right parenthesis	Rechte Klammer zu viel

ERKLÄRUNG:
Sie haben für ein Overlaymodul, das in Klammern eingegeben werden muß, die linke Klammer vergessen. Das folgende Modul wird ebenfalls in Klammern gesetzt. Korrigieren Sie die Eingabezeile.

Fehlermeldungen

No object modules specified	Keine Objektmodule benannt

ERKLÄRUNG:
Sie haben LINK ohne ein Objektmodul aufgerufen. Damit kann das Programm nicht arbeiten. Benennen Sie mindestens ein Modul.

Program size or number of segments exceeds capacity of Linker	Programmgröße oder Anzahl der Segmente übersteigt die Kapazität von LINK

ERKLÄRUNG:
LINK kann ein Programm von maximal 1 MB Größe mit maximal 1024 Segmenten und Klassen erstellen. Teilen Sie das Programm.

Segment limit set too high	Segmentgrenze zu hoch

ERKLÄRUNG:
Ein Eingabemodul enthält falsche Segmentwerte. Prüfen Sie das Modul.

Stack size exceeds 64KB	Stapelsegment ist größer als 64KB

ERKLÄRUNG:
Das Stapelsegment darf nur bis zu 64 KB groß sein. Starten Sie LINK neu, und bestimmen Sie mit dem Zusatz /STACK ein kleineres Segment.

Segment size exceeds 64KB	Segment ist größer als 64KB

ERKLÄRUNG:
Das Segment darf nur 64KB betragen, damit es noch adressiert werden kann. Überprüfen Sie das Modul.

Symbol already defined	Symbol mehr als einmal definiert

ERKLÄRUNG:
Von LINK wurden zwei oder mehr Definitionen für ein Symbol gefunden. Ändern Sie ein Modul, und starten Sie LINK neu.

Symbol table capacity exceeded	Speicherkapazität der Symboltabelle überschritten

ERKLÄRUNG:
Durch zu viele oder zu lange Namen ist der Speicher der Tabelle (ca. 25KB) voll. Geben Sie kürzere Namen ein.

Too many external symbols in one module	Zu viele externe Symbole in einem Modul

ERKLÄRUNG:
Es sind maximal 1024 externe Symbole in einem Modul erlaubt. Ändern Sie das Modul, und starten Sie LINK neu.

Too many groups	Zu viele Gruppen

ERKLÄRUNG:
Es sind maximal 10 Gruppen erlaubt. Ändern Sie das Objektmodul, und starten Sie LINK neu.

Too many libraries specified	Zu viele Bibliotheken benannt

ERKLÄRUNG:
Es sind maximal sechzehn LIB-Dateien erlaubt. Starten Sie LINK neu, geben Sie weniger Dateien ein, oder bestimmen Sie mit der Variablen LIB ein Verzeichnis.

Too many overlays	Zu viele Overlaymodule

ERKLÄRUNG:
Sie haben in der Eingabezeile zu viele Module als Overlay gekennzeichnet.

Too many PUBLIC symbols	Zu viele PUBLIC-Symbole

ERKLÄRUNG:
Es sind maximal 1024 Symbole erlaubt. Ändern Sie die Objektmodule, und starten Sie LINK neu.

Fehlermeldungen

Too many segments or classes	Zu viele Segmente oder Klassen

ERKLÄRUNG:
Die maximale Zahl von 255 Segmenten und Klassen zusammen wird überschritten. Ändern Sie die Objektmodule, und starten Sie LINK neu.

Unresolved externals: list	Ungelöste Querreferenzen

ERKLÄRUNG:
Es folgt eine Liste der externen Symbole, welche kein erklärendes Modul enthalten. Überprüfen Sie die PUBLIC- und externen Symbole, und ändern Sie die Module.

VM read error	Lesefehler der Datei VM.TMP

ERKLÄRUNG:
Die Diskette mit der Datei VM.TMP ist beschädigt. Tauschen Sie die Diskette.

Warning: No STACK segment	Warnung: Kein Stapelsegment

ERKLÄRUNG:
Keines der Objektmodule enthält eine Größenangabe zum Stapelsegment. Sie haben den Zusatz **/STACK** nicht eingegeben. Ändern Sie das Objektmodul, oder rufen Sie LINK mit /STACK-Zusatz auf. Wollen Sie eine COM-Datei mit dem Programm EXE2BIN erstellen, oder bestimmt das neue Programm das Stapelsegment selber, können Sie diese Warnung ignorieren.

Kapitel 15

EXE2BIN

Übersicht

Das Dienstprogramm EXE2BIN formt Dateien des Formates .EXE in Dateien des Formates .COM um. Das Programm ist auf der ersten Systemdiskette enthalten.

In diesem Kapitel wird vorausgesetzt, daß dem Leser der Maschinenkode der Prozessoren 8086 und 80286 der Fa. Intel vertraut ist. Ebenso sollten Begriffe wie Bit, Byte, Flag, Register und Segment bekannt sein.

Das Programm EXE2BIN (ausgesprochen EXE to BIN) formt Programme des Formates EXE (ausführbare Programme), die das Programm LINK erstellt hat, in reine Binärdateien um. Bei entsprechender Programmierung können die Dateien mit dem Zusatz COM versehen werden. Sie werden dann vom Betriebssystem ebenfalls als ausführbare Programme geladen. Da das Lademodul im Kopf der Datei fehlt, sind sie kleiner und können schneller geladen werden. Ebenso können sie ohne Beachtung von Speicheradressen an jede beliebige Stelle des Arbeitsspeichers geladen werden.

Programmaufruf

Zweck

formt Dateien mit dem Format .EXE in das Format .COM um.

Syntax

[d:] [Weg] EXE2BIN Dateibez [D: [Dateiname. [Zus]]]

dabei bedeutet:

d:	vor dem Programmnamen bestimmt das aufzurufende Laufwerk
Weg	vor dem Programmnamen bestimmt das aufzurufende Verzeichnis
Dateibez	Name der Datei, die umgeformt werden soll
D:	Laufwerk, auf das die neue Datei geschrieben werden soll
Dateiname	Name der neuen Datei, ohne Eingabe wird der vorherige Name der .EXE-Datei genommen
Zus	Namenszusatz der neuen Datei, ohne Eingabe wird .BIN angenommen

Bemerkungen

Dieses Dienstprogramm formt Dateien mit einem gültigen .EXE-Format, die eine maximale Größe von 64 Kb haben, zu reinen *Binärdateien* um. Diese Dateien haben dann das Format der .COM-Dateien. Sie enthalten keine Speicherplatzbestimmung mehr und sind im Arbeitsspeicher frei verschiebbar. Die EXE-Datei muß vom Dienstprogramm LINK erstellt worden sein und die Zeichen „MZ" im Programmkopf enthalten.

Programmaufruf

Geben Sie für die umzuformende Datei keinen Zusatz mit ein, wird der Zusatz .EXE angenommen. Die neue Datei wird auf das bezeichnete Laufwerk mit dem in der Befehlszeile angegebenen Namen geschrieben. Geben Sie dies nicht an, wird die neue Datei mit dem vorherigen Namen und dem Zusatz .BIN auf das Arbeitslaufwerk geschrieben.

Die umzuformende Datei darf im Programmkopf keine Angabe für das Stapelsegment enthalten. Abhängig von den Bestimmungen für das CS-Segment und den Programmzähler sind zwei Arten der Umformung möglich:

1. Werden die Werte für den Programmanfang (Register CS und IP) nicht definiert, wird eine reine *Binärumformung* vorgenommen. Enthält das Programm Anweisungen für Speicherplatzbestimmungen, werden Sie aufgefordert, diese einzugeben. Das ist dann die absolute Adresse im Arbeitsspeicher, an die das Programm geladen wird. Das Programm kann dann nur korrekt abgearbeitet werden, wenn es an diesen Platz geladen wird. Dies kann nur durch ein Anwenderprogramm vorgenommen werden. Das Betriebssystem ist dazu nicht in der Lage.
2. Werden die Werte für das Segmentregister (Register CS) mit 0000h und für den Programmzähler (Register IP) mit 100h definiert, wird die Datei zu einer *normalen .COM-Datei* umgeformt. Die Startadresse der Datei ist dann 100h, die ersten 100h Bytes der Datei werden gelöscht. Dies kann durch die Assembleranweisung ORG bestimmt werden. In diesem Falle dürfen keine Speicherplatzbestimmungen vorgenommen werden. .COM-Dateien müssen frei verschiebbar sein. Nach der Umformung kann die neue Datei mit dem Zusatz .COM versehen werden. Sie wird dann wie jede andere .COM-Datei behandelt.

Enthält die Datei Angaben über den Speicherplatz, werden Sie während der Umformung aufgefordert, die Adresse einzugeben. Es erscheint folgende Aufforderung:

Fixups needed base-segment (hex):	Speicherplatzbestimmung erforderlich - Adresse (hex):

Geben Sie in hexadezimaler Schreibweise das erforderliche Segment ein. Die Datei arbeitet später nur noch dann korrekt, wenn sie ab der Adresse geladen wird. Dies muß durch ein entsprechendes Programm erfolgen. Die Datei COMMAND.COM kann dies nicht.

Fehlermeldungen

File cannot be converted	Datei kann nicht umgeformt werden

ERKLÄRUNG:
Die Quelldatei kann nicht umgeformt werden, da sie nicht das richtige Format hat. Entweder sind die Werte für CS und IP nicht richtig definiert, oder es ist ein Stapelsegment vorhanden.

File not found	Datei nicht vorhanden

ERKLÄRUNG:
Die umzuformende Datei ist nicht auf der Platte.

Insufficient memory	Arbeitsspeicher zu klein

ERKLÄRUNG:
Der freie Arbeitsspeicher ist zu klein. Das Programm EXE2BIN kann nicht korrekt arbeiten.

File creation error	Falsche Dateianlage

ERKLÄRUNG:
Im Inhaltsverzeichnis ist kein Platz mehr, um eine weitere Datei zu speichern. Sofern noch freier Speicherplatz auf der Platte vorhanden ist, müssen im Verzeichnis Einträge gelöscht werden, damit eine weitere Datei gespeichert werden kann.

Fehlermeldungen

Insufficient disk space	Nicht genug Speicherkapazität auf der Platte

ERKLÄRUNG:
Die Speicherkapazität der Platte, auf die geschrieben werden soll, reicht nicht aus für die Datei. Wechseln Sie das Laufwerk.

WARNING - Read error on EXE file. Amount read less than size in header	WARNUNG- Lesefehler der EXE-Datei. Die tatsächliche Größe ist geringer als im Kopf der Datei angegeben

ERKLÄRUNG:
Diese Meldung ist nur eine Warnung. Die Datei auf der Platte ist geringer als die im Kopf angegebene Größe. Die Datei könnte fehlerhaft sein. Erstellen Sie die Datei neu.

DEBUG

Übersicht

Das MS-DOS Dienstprogramm DEBUG ist ein *Testprogramm* für compilierte binäre Dateien. Es ist ein Werkzeug zur Programmherstellung, das in den Arbeitsspeicher geladene Programme anschließend testet. Das Programm finden Sie auf der ersten Systemdiskette.

In diesem Kapitel wird vorausgesetzt, daß der Leser sich mit dem Maschinencode der Prozessoren 80286 und 8086 der Fa. Intel auskennt. Ebenso sollte der Leser Programmiererfahrung haben und Begriffe wie Bit, Byte, Flag, Register und Segment kennen.

Das *Monitorprogramm MFM-200* enthält ebenfalls ein Testprogramm, das dem Programm DEBUG sehr gleicht. Funktion und Arbeitsweise sind bis auf die Befehle ASSEMBLE, PROCEED und WRITE identisch. Sie können dieses Monitortestprogramm jederzeit durch die Tastenkombination **CTRL-ALT-RETURN** aufrufen. Das laufende Programm wird sofort unterbrochen. Verlassen Sie das Monitorprogramm mit dem Befehl **G**(o), kehren Sie zur Stelle der Programmunterbrechung zurück. Das laufende Programm wird dann normal weiter abgearbeitet.

Programmaufruf

[d:] [Weg] DEBUG Dateibez Variable < Eingabe

dabei bedeutet:

d:	vor dem Programmnamen bestimmt das aufzurufende Laufwerk
Weg	vor dem Programmnamen bestimmt das aufzurufende Verzeichnis
Dateibez	Name der zu testenden Datei mit Laufwerkbezeichnung und/oder Verzeichnisname
Variable	Zusatzeingabe für das zu testende Programm
< Eingabe	Textdatei mit gespeicherten DEBUG-Befehlen

Bemerkungen

Das Programm DEBUG ist das Gegenstück zum Dienstprogramm EDLIN. Das Programm EDLIN erstellt oder ändert Textdateien. Das Programm DEBUG erstellt oder ändert binäre Dateien.

Mit Hilfe des Programmes DEBUG haben Sie die Möglichkeit, schnell einen Fehler zu finden und zu ändern. Sie brauchen für die Fehlersuche das Programm nicht zu disassemblieren. Sie können eine Speicheradresse oder ein Register der CPU direkt ändern. Danach kann das geänderte Programm sofort getestet werden. Ist es jetzt fehlerfrei, kann es abgespeichert werden.

Sämtliche Befehle des Programmes können jederzeit durch die Tastenkombination **CTRL-C** abgebrochen werden. Die Bildschirmausgabe kann durch die Tastenkombination **CTRL-S** angehalten werden. Ebenso können Sie sämtliche Funktionstasten zum Korrigieren der Eingabe benutzen.

Das Programm DEBUG wird zusammen mit dem Namen der zu testenden Datei aufgerufen. Dabei ist es möglich, Variable mit einzugeben. Nach dem Programmnamen wird erst die zu testende Datei eingegeben, und dann, falls gewünscht, Variable. Dabei werden alle Zusatzeingaben der Testdatei ab der Adresse 80h des PSP (Programm Segment Prefix) übergeben. Damit erfolgt der Aufruf der Testdatei in der Weise wie auch mit dem Betriebssystem.

Programmaufruf

Mit dem Aufruf von DEBUG wird der *Programmkopf* an die relative Adresse 0h des DS-Registers im Arbeitsspeicher gesetzt. Die vorherigen Versionen erlaubten ein Überschreiben dieses Kopfes. Dies ist jetzt nur noch gestattet, wenn keine zu testende Datei aufgerufen wird. Bearbeiten Sie eine .COM- oder .EXE-Datei, sollten Sie niemals unterhalb der relativen Adresse 5Ch arbeiten. Dies könnte zum Abbruch des Programmes führen.

Wurde ein Programm mit der Meldung

```
Programm terminated normaly -        Programm normal beendet
```

beendet, muß es mit den Befehlen **N**(ame) und **L**(oad) neu gestartet werden.

Soll ein Programm mit DEBUG zum Testen aufgerufen werden, z.B.:

```
DEBUG TEST.COM
```

wird die Datei *TEST.COM* im Arbeitsspeicher in das niedrigste freie Segment geladen. Die Register BX:CX enthalten die Anzahl der geladenen Bytes.

Das Programm meldet sich anschließend mit seinem *Prompt*, einem Bindestrich (-).

Die Datei TEST.COM wird so in den Arbeitsspeicher geladen, wie sie auf Betriebssystemebene mit einer normalen Befehlszeile aufgerufen worden wäre.

Wird das Programm DEBUG ohne Dateinamen aufgerufen, oder die benannte Datei nicht gefunden, erscheint sofort das Prompt, und Sie können Befehle eingeben. Es ist jetzt möglich, den Arbeitsspeicher zu untersuchen, absolute Diskettensektoren oder aber auch eine zu testende Datei nachzuladen.

Befehle

Jeder *Befehl* des Programmes DEBUG besteht aus einem einzelnen Buchstaben sowie *wahlweise einzugebenden Werten.* Die Tabelle 16.1 zeigt eine Übersicht der Befehle mit ihrer Schreibweise. Die Tabelle 16.2 erklärt die zusätzlich einzugebenden Werte, die die Arbeitsweise des Befehles näher bestimmen. Dabei können sämtliche Funktions- und Kontrolltasten benutzt werden (s. a. Kapitel 5). Die Eingabe kann in Groß- oder Kleinbuchstaben erfolgen, ebenso können *Trennungszeichen* (Leertaste oder Komma) eingegeben werden. Erforderlich sind Trennungszeichen immer zwischen zwei hexadezimalen Werten. Die beiden folgenden Eingaben

DCS:0100 110 und D CS:0100 110

sind gleich.

Das Programm DEBUG liest die Standardeingabe des Betriebssystemes. Damit kann sehr einfach die Funktion der Datenumleitung benutzt werden. Speichern Sie alle auszuführenden Befehle des Programmes in einer Textdatei. Teilen Sie dann dem Programm beim Aufruf in der Befehlszeile durch das Zeichen „<" mit, daß die folgende Datei alle Befehle enthält. Sie können damit Befehlsmakros abarbeiten lassen. Vergessen Sie jedoch nie, als letzten Befehl Q(uit) einzugeben. Bei Datenumleitung ist keine Eingabe über die Tastatur mehr möglich.

Tabelle 16.1 DEBUG-Befehle

NAME	SCHREIBWEISE	
Assemble	A	[Adresse]
Compare	C	Bereich Adresse
Dump	D	[Adresse] [L Wert]
	D	[Bereich]
Enter	E	Adresse [Liste]
Fill	F	Bereich Liste
Go	G	[=[Adresse1] [Adresse2...]]
Hex	H	Wert Wert
Input	I	Wert Byte
Load	L	[Adresse [Laufwerk Sektor Anzahl]]
Move	M	Bereich Adresse
Name	N	Dateibez [Dateibez]
Output	O	Wert Byte
Proceed	P	[=[Adresse] [Wert]]
Quit	Q	
Register	R	[Register]
Search	S	Bereich Liste
Trace	T	[=[Adresse] [Wert]]
Unassemble	U	[Adresse [L Wert]]
	U	[Bereich]
Write	W	[Adresse [Laufwerk Sektor Anzahl]]

Bei falscher Schreibweise der Befehle wird der erkannte Fehler durch einen *Pfeil* (ˆ) angezeigt. Ein Beispiel:

```
DCS:100 CS:110
        ˆError
```

zeigt an, daß der Bereich falsch eingegeben wurde. Er darf nicht durch einen Registerwert bestimmt werden.

Befehle

Tabelle 16.2 DEBUG-Variable

VARIABLE	ERKLÄRUNG
Adresse	Eine einteilige Angabe, bestehend aus einem Relativwert. Der Befehl **D**(ump) bezieht sich dann auf das DS-Segment, die Befehle **E**(nter), **F**(ill), **G**(o), **P**(roceed), **T**(race) und **U**(nassemble) beziehen sich auf das CS-Segment. Eine zweiteilige Angabe, bestehend aus: ● einer alphabetischen *Segmentregisterangabe* ● einer vierstelligen *Segmentadresse* mit einem *Relativwert.* Werden diese Werte nicht eingegeben, wird mit den vom Programm vorgegebenen Werten gearbeitet. Alle numerischen Werte sind hexadezimal. Ein Doppelpunkt (:) muß zwischen eine Segmentadresse und einen Relativwert gesetzt werden. Ein Beispiel: CS:0100 04BA:0100
Byte	Eingabe in ein Register oder in eine Adresse von *zwei Zeichen* mit hexadezimalem Wert
Laufwerk	*Zahl* mit hexadezimalem Wert zur Bezeichnung des Laufwerkes, auf das geschrieben oder gelesen werden soll. Gültige Werte sind: 0=A:, 1=B:, 2=C:, 3=D:, 4=E:, 5=F:, 6=G:, 7=H
Liste	Eingabe von einem oder mehreren *Bytes*, muß immer als letztes in der Befehlszeile eingegeben werden. Ein Beispiel: F CS:100 42 45 52 54 41

Tabelle 16.2 DEBUG-Variable

VARIABLE	ERKLÄRUNG
Bereich	Eingabe von *Anfangsadresse, Endadresse*; bezeichnet den Bereich, in dem der Befehl abgearbeitet werden soll. Ein Beispiel: CS:100,110 Es ist auch möglich, als zweiten Wert eine Anzahl von Werten einzugeben. Ein Beispiel: CS:100 L 10 Es werden 10 Werte ab der relativen Adresse 100h des CS-Segmentes bestimmt. Der Bereich darf maximal 10000h betragen, die Eingabe hierfür: **0000** oder **0**.
Sektor	Eine Zahl mit *hexadezimalem Wert*, um die logische Satznummer auf der Diskette zu definieren.
Register	Eine *zweistellige alphabetische Bezeichnung* für eine spezielle Speichereinheit, wo kurzfristig Informationen gespeichert werden können; z. B. arithmetische Operanden oder Speicheradressen.
Zeichenkette	besteht aus *mehreren Zeichen*, eingeschlossen in einfache (') oder doppelte (") Anführungsstriche. Werden die Zeichen im Text selber benutzt, müssen Sie die Zeichenkette doppelt einschließen. Ein Beispiel: 'Diese "Zeile" ist richtig.' "Diese 'Zeile' ist richtig." "Diese ""Zeile"" ist richtig." "Diese "Zeile" ist falsch."

Befehle

Tabelle 16.2 DEBUG-Variable

VARIABLE	ERKLÄRUNG
Wert	Eine *Zahl* mit hexadezimalem Wert. Damit wird z. B. ein Ein- oder Ausgabekanal definiert oder die Anzahl bestimmt, wie oft der Befehl auszuführen ist.

Auf den folgenden Seiten werden alle Befehle ausführlich mit kurzen Beispielen beschrieben.

Assemble

Syntax

A [Adresse]

Zweck

schreibt Mnemonics der Prozessoren 8086/8087/8088 direkt in den Arbeitsspeicher.

Bemerkungen

Alle numerischen Werte müssen mit gültigen Hexadezimalzahlen eingegeben werden. Wird ein Eingabefehler festgestellt, erscheint das Fehlerzeichen (ˆ) unter dem falschen Befehl.

Präfixe müssen vor dem Prozessorbefehl, auf den sie sich beziehen, eingegeben werden. Sie können in derselben Zeile oder einer separaten Zeile eingegeben werden. Es gibt folgende Präfixe:

REP	wiederhole
REPZ	wiederhole während Null
REPE	wiederhole während gleich
REPNZ	wiederhole während nicht Null
REPNE	wiederhole während nicht gleich

Die *Präfix-Mnemonics* zum Überschreiben der Segmente sind:

CS: DS: ES: SS:

Diese Präfixe müssen immer in einer separaten Zeile vor dem Befehl eingegeben werden.

Bei *Mnemonics* zur Stringmanipulation muß immer die Stringgröße definiert werden; z. B. MOVSW für Wort-Strings (2 Bytes) oder MOVSB für Byte-Strings.

Das *Mnemonic* für einen fernen Rücksprung ist RETF.

Assemble

Der Assembler setzt automatisch Sprungadressen als kurz, nah oder fern. Das kann mit dem Präfix NEAR (NE) oder dem Präfix FAR überschrieben werden. Mit diesen Präfixen wird ein Zielort bezeichnet. Ein Beispiel:

```
0100:0500  JMP   502        ; ein 2-Byte-Sprung, kurz
0100:0502  JMP   NEAR 505   ; ein 3-Byte-Sprung, nah
0100:0505  JMP   FAR 50A    ; ein 5-Byte-Sprung, fern
```

Das Programm erkennt nicht, ob sich ein Operand auf einen *Wortspeicherplatz* oder einen *Bytespeicherplatz* bezieht. Daher muß der Datentyp mit dem Präfix WORD PTR (WO) oder BYTE PTR (BY) definiert werden. Ein Beispiel:

```
Neg     BYTE PTR [128]    ; negiere den Speicherplatz
DEC     WO PTR [SI]       ; vermindere den Speicherplatz um 1
```

Ebenso kann das Programm nicht erkennen, ob sich der *Operand* auf einen Speicherplatz oder eine absolute Zahl bezieht. Das Programm setzt daher den Wert des Speicherplatzes in Klammern. Ein Beispiel:

```
MOV     AX,21             ; lade AX mit 21h
MOV     AX,[21]           ; lade AX mit dem Wert des Speicher-
                          ; platzes 21h
```

Der Operationskode DB schreibt *Byte-Werte* direkt in den Speicher, der Operationskode DW schreibt *Wort-Werte* direkt in den Speicher. Ein Beispiel:

```
DB      1,2,3,4, "Dies ist ein Beispiel"
DW      1000,2000,3000,"BACH"
```

Alle Formen der Adressberechnung werden verarbeitet. Ein Beispiel:

```
ADD     BX,34 [BP+2].[SI-1]
POP     [BP+DI]
PUSH    [SI]
```

Für den Prozessor 8087 müssen die *Befehle* WAIT und FWAIT genau definiert werden. Ein Beispiel:

```
FWAIT FADD ST,ST(3)       ; diese Zeile ist richtig
FLD TBYTE PTR ÄBXÜ        ; diese Zeile nicht
```

Compare

Syntax

C Bereich Adresse

Zweck

vergleicht zwei Bereiche des Speichers.

Bemerkungen

Sind beide Bereiche identisch, erfolgt keine Ausgabe, und es erscheint wieder das Prompt. Unterschiede werden in der Form

Adresse1 Byte1 Byte2 Adresse2

ausgegeben. Ein Beispiel

C0:0 L12 CS:51

vergleicht den Speicher von Adresse 0h bis 11h (L12 = 18 Speicherstellen) mit dem Speicher ab der relativen Adresse 51h des CS-Registers. Auf dem Bildschirm erscheint folgende Anzeige:

```
0000:0000  70  21  09C3:0051
0000:0001  2A  CB  09C3:0052
0000:0002  19  00  09C3:0053
0000:0003  01  00  09C3:0054
0000:0004  6C  00  09C3:0055
0000:0005  01  00  09C3:0056
0000:0006  70  00  09C3:0057
0000:0008  BB  00  09C3:0059
0000:0009  03  00  09C3:005A
0000:000B  FC  00  09C3:005C
0000:000C  6C  20  09C3:005D
0000:000D  01  20  09C3:005E
0000:000E  70  20  09C3:005F
0000:000F  00  20  09C3:0060
0000:0010  6C  20  09C3:0061
0000:0011  01  20  09C3:0062
```

Dump

Syntax

D [Adresse [L Wert]]
D [Bereich]

Zweck

zeigt ab der eingegebenen *Adresse* den Speicherinhalt.

Bemerkungen

Bei Eingabe nur einer Adresse werden 128 Bytes angezeigt. Geben Sie einen Bereich ein, wird dieser Bereich angezeigt. Wird der Befehl ohne Zusatz eingegeben, erfolgt die Speicheranzeige beginnend ab dem Wert des DS-Registers.

Die Ausgabe erfolgt in drei Teilen:

- einem *Adressteil*; mit der Segmentangabe und dem jeweiligen Relativwert.
- einem *hexadezimalen Teil*; jeder Speicherplatz wird mit seinem Hexadezimalwert angezeigt.
- einem *ASCII-Teil*; jeder Speicherplatz wird mit einem ASCII-Zeichen dargestellt.

Im ASCII-Teil werden nicht darstellbare Zeichen als Punkt angezeigt. Jede Zeile zeigt 16 Bytes, beginnend an einem 16-Byte-Grenzwert (Paragraph). Zwischen dem achten und neunten Byte wird zur besseren Übersicht ein Bindestrich eingefügt. Ist die eingegebene Adresse kein 16-Byte-Grenzwert, wird nur der restliche Teil in der ersten Zeile gezeigt, ebenso wie in der letzten Zeile. Die komplette Anzeige beträgt immer 128 Bytes.

Zum Beispiel könnte nach Eingabe von

```
D CS:100
```

folgendes angezeigt werden:

```
0D8B:0100  A4 1F 26 A3 0E 00 FC A3-08 0F A2 3C 07 E8 45 F8 $.&#..ö#..“<.hEX
0D8B:0110  BE 80 00 AC 8A C8 32 ED-E3 43 BE 81 00 AC 3C 20 >..,.H2mcC>..,<
0D8B:0120  74 17 3C 09 74 13 3A 06-AA 0A 75 34 49 E3 29 AC t.<.t.:.*.u4IC),
0D8B:0130  0C 20 3C 70 75 03 E9 03-01 3C 64 75 09 C6 06 18 . <pu.i..<du.F..
0D8B:0140  0F 01 E2 D9 EB 12 3C 63-75 11 89 36 6C 08 C6 06 ..bYk.<cu..6l.F.
0D8B:0150  6B 08 00 C6 06 18 0F 01-E9 F3 00 E2 C0 E9 1C 01 k..F....is.bi..
0D8B:0160  4E 8B D6 51 56 AC 49 3C-20 74 09 3C 09 74 05 E3 N.VQV,I< t.<.t.c
0D8B:0170  02 EB F2 46 C6 44 FF 00-B8 02 3D CD 21 72 46 8B .krFFD..8.=M!rF.
```

Geben Sie den Befehl ohne Adressangabe erneut ein, wird als neuer Anfangswert der zuletzt ausgegebene Endwert genommen.

Enter

Syntax

E Adresse [Liste]

Zweck

Zeigt die *Adresse* mit dem aktuellen Inhalt und wartet auf eine Eingabe.

Bemerkungen

Erfolgt mit der Eingabe des Befehls außer der Adresse auch der neue *Wert*, wird dieser Wert automatisch in den Speicher geschrieben. Dabei können Sie Buchstaben oder Worte direkt eingeben. Schließen Sie die Eingabe dann in Anführungsstriche ein. Geben Sie einen ungültigen Wert ein, ändert sich nichts. Geben Sie nur die Adresse ein, wird die Adresse mit dem Wert wiederholt. Der Cursor steht hinter dem Wert und zeigt an, daß hier eine Eingabe erwartet wird. Sie können jetzt folgendermaßen einen neuen Wert eingeben:

1. Geben Sie nach dem gezeigten Wert den neuen Wert ein.

   Es werden nur gültige Hexadezimalwerte von 00h bis FFh akzeptiert.

2. Mit Betätigen der Leertaste gelangen Sie zum nächsten Byte. Auch hier wird erst wieder der alte Wert angezeigt. Geben Sie den neuen Wert ein, wie unter 1 beschrieben.

   Nach jedem achten Byte wird dabei in einer neuen Zeile angefangen.

3. Mit Eingabe eines Bindestriches (-) gelangen Sie um je ein Byte zurück. Dabei wird auf dem Bildschirm jeweils in einer neuen Zeile angefangen.

4. Schließen Sie die Eingabe mit **RETURN** ab.

Ein Beispiel:

```
E CS:100
```

zeigt folgendes:

```
04BA:0100 EB._
```

Es ist jetzt möglich, den Wert zu verändern, z. B. auf **41**.

```
04BA:0100 EB.41
```

Nach zweimaligem Betätigen der Leertaste erscheinen die nächsten zwei Werte, die nicht verändert werden sollen.

```
04BA:0100 EB.41 10. 00. BC._
```

Der Wert BC kann jetzt auf **42** geändert werden:

```
04BA:0100 EB.41 10. 00. BC.42
```

Stellen Sie jetzt fest, daß der Wert 10 auf F6 geändert werden muß, geben Sie zweimal einen Bindestrich ein und danach **F6**:

```
04BA:0100 EB.41 10. 00. BC.42
04BA:0102 00.-
04BA:0101 10.F6
```

Mit **RETURN** kehren Sie zur Befehlszeile zurück.

Fill

Syntax

F Bereich Liste

Zweck

gibt die Werte der *Liste* in die Adressen des Bereiches ein.

Bemerkungen

Enthält die Liste weniger Bytes als der bezeichnete Bereich, wird die *Liste* solange wiederholt, bis der Bereich gefüllt ist. Enthält die Liste mehr Bytes als der Bereich umfaßt, werden die zusätzlichen Bytes ignoriert. Ist eine Adresse im angegebenen Bereich nicht korrekt, wird der Fehler auch für die folgenden Adressen übernommen.

Die Funktion F enthält die Funktion E mehrmals. Damit können mehrere Speicherplätze mit einer Funktion geändert werden. Ein Beispiel:

```
F04BA:100 L 100 42 45 52 54 41
```

Es wird jetzt der Speicher von 04BA:100 bis 04BA:1FF mit den fünf Bytes gefüllt. Dabei werden die fünf Bytes solange wieder eingelesen, bis alle 100h Bytes gefüllt sind.

Go

Syntax

G [=[Adresse1 [Adresse2...]]

Zweck

führt das Programm im Arbeitsspeicher bis zu einer *Stopadresse* aus.

Bemerkungen

Erfolgt die Eingabe des Befehls ohne Angabe der Adresse, wird das Programm so ausgeführt, wie es normalerweise vom Betriebssystem aus geschehen wäre.

Wird die *Adresse1* eingegeben, beginnt die Programmausführung ab dieser Adresse. Fehlt die Segmentangabe bei der *Adresse1*, wird nur der Programmzähler (Register IP) gesetzt. Geben Sie die Segmentadresse mit ein, werden sowohl das CS-Segment als auch der Programmzähler gesetzt. Die Eingabe des Gleichheitzeichens (=) ist erforderlich, um die Start- (Adresse1) von der Stopadresse (Adresse2) zu unterscheiden.

Sind *Stopadressen* eingegeben, hält das Programm an, sobald es auf eine der bezeichneten *Stopadressen* trifft. Es werden dann sämtliche Register und Flags sowie die Instruktion, die als nächstes abgearbeitet werden soll, angezeigt.

Maximal sind 10 Stopadressen möglich. Diese müssen immer auf das erste Byte eines Prozessorbefehles zeigen. Werden mehr Adressen eingegeben, erscheint die Fehlermeldung:

BP (Too many breakpoints)	Zu viele Stopadressen

Go

Der Stapelzeiger (Register SP) muß auf einen gültigen Bereich zeigen und mindestens noch sechs Bytes speichern können. Der Befehl G benutzt ein IRET (Interrupt RETurn) zur Programmausführung. Der Stapelzeiger wird gesetzt, die Register CS, IP und das Statusregister werden auf den Stapel übertragen.

(Ist der Stapel zu klein, kann die Funktion nicht ausgeführt werden.) Ein 1-Byte-Unterbrechungsvektor (CCh) wird an die Stopadressen gesetzt. Gelangt das Programm zur Stopadresse, werden alle Stopadressen wieder mit ihren alten Werten versehen. Hält das Programm nicht an einer Stopadresse, werden die Adressen mit den Unterbrechungsvektoren nicht durch ihre alten Werte ersetzt.

Ein Beispiel:

```
G CS:7550
```

Das Programm wird bis zur Adresse 7550 im CS-Segment abgearbeitet. Danach werden sämtliche Registerinhalte angezeigt, und der Befehl ist beendet.

Geben Sie nach diesem Stop den Befehl **G** neu ein, wird das Programm ab der Stopadresse weiter abgearbeitet.

Hex

Syntax

H Wert Wert

Zweck

rechnet hexadezimal

Bemerkungen

Geben Sie zwei *hexadezimale Werte* ein. Zuerst werden die beiden Werte addiert, dann subtrahiert. Das Ergebnis wird in der folgenden Zeile angezeigt. Ein Beispiel:

H10A 19F

Das Ergebnis lautet:

02A9 FF6B

Input

Syntax

I Wert

Zweck

liest und zeigt den Wert eines Ein- / Ausgabekanals.

Bemerkungen

Geben Sie den Wert hexadezimal ein. Eine 16-Bit-Adresse ist erlaubt. Ein Beispiel:

I2F8

Ist der aktuelle Wert dieses Kanals z. B. 42h, wird vom Programm folgendes angezeigt:

42

Load

Syntax

L [Adresse [Laufwerk Sektor Anzahl]]

Zweck

lädt Dateien oder Sektoren in den Arbeitsspeicher

Bemerkungen

Die Register BX:CX zeigen die Zahl der geladenen Bytes. Werden mit diesem Befehl Dateien geladen, müssen diese vorher beim Programmaufruf oder durch den Befehl N(ame) definiert worden sein. In beiden Fällen wird ein Dateiname, wie er im normalen FCB ab DS:5C vorhanden ist, erzeugt.

Bei der Eingabe des Befehls L ohne weitere Angaben wird die Datei ab der Adresse CS:100 in den Speicher geladen. Dies gilt nicht für .EXE-Dateien. Die Register BX:CX zeigen dabei immer die Zahl der geladenen Bytes an.

Bei der Angabe einer Adresse wird die Datei ab dieser Adresse in den Speicher geladen. Dies gilt nicht für Dateien mit dem Zusatz .COM oder .EXE. Diese Dateien werden immer ab der Adresse DS:100 geladen. Zusätzlich werden bei .EXE-Dateien die Register CS, IP, SS und SP mit den Werten des Dateikopfes geladen.

Geben Sie sämtliche Variable ein, werden absolute Diskettensektoren geladen. Dabei wird das Laufwerk folgendermaßen bezeichnet:

0=A:, 1=B:, 2=C:, 3=D:, 4=E:, 5=F:, 6=G:, 7=H:.

Es wird dann ab dem ersten angegebenen absoluten Sektor geladen.

Für das folgende Beispiel wird die Datei FILE.COM nachgeladen. Rufen Sie das Programm auf:

DEBUG

Load

Bezeichnen Sie die nachzuladende Datei:

```
N FILE.COM
```

Laden Sie die Datei durch Eingabe von:

```
L
```

DEBUG lädt die Datei in den Speicher und meldet sich wieder mit dem Prompt (-).

Wollen Sie nur einen Teil einer Datei von der Platte laden, geben Sie ein:

```
L 04BA:100 2 0F 6D
```

Jetzt werden 109 (6Dh) Sektoren vom Laufwerk C: (2h), beginnend ab dem Sektor 15 (0Fh), geladen.

Erfolgt der Aufruf einer Datei mit dem Zusatz .EXE, wird sie an die im Kopf der Datei benannte Adresse geladen. Ihre Adresseingabe wird dabei vom Programm immer ignoriert. Dabei wird der Kopf der Datei nicht miterfaßt. Daher ist eine .EXE-Datei unter DEBUG immer kleiner als auf Platte.

Move

Syntax

M Bereich Adresse

Zweck

verschiebt einen Speicherblock an die bezeichnete Adresse.

Bemerkungen

Dieser Befehl verschiebt Daten im Arbeitsspeicher, indem er die Daten kopiert. Dabei werden Verschiebungen mit überlappenden Blöcken immer ohne Verlust von Daten ausgeführt. Es werden die möglichen Adressen zuerst kopiert. Verschieben Sie Daten zu einer niedrigeren Speicheradresse, werden zuerst die Daten im unteren Bereich verschoben. Verschieben Sie Daten zu einer höheren Adresse, wird zuerst der obere Bereich verschoben.

Ein Beispiel:

M CS:100 110 CS:500

Zuerst wird der Wert von CS:110 nach CS:510 übertragen, danach CS:10F nach CS:50F usw. Mit dem Befehl D läßt sich anschließend das Ergebnis leicht überprüfen.

Name

Syntax

N Dateibez1 [Dateibez2]

Zweck

definiert einen Dateinamen

Bemerkungen

Mit dem Befehl **N**(ame) können maximal zwei Namen eingegeben werden. Dabei darf den Namen eine Laufwerkbezeichnung und/ oder ein Verzeichnisname vorangestellt werden.

Der Befehl **N** hat zwei Funktionen:

1. Zunächst findet eine Definition der später zu ladenden oder zu schreibenden Datei statt. Daher ist es möglich, das Dienstprogramm DEBUG ohne einen Dateinamen aufzurufen. Ist die Datei später nachzuladen, wird mit dem Befehl N der Name der Datei festgelegt.
2. Sie definieren Variable, die dem angegebenen Dateinamen zuzuordnen sind.

Ein Beispiel:

```
N FILE1.EXE
L
G
```

Auf Grund der zwei Funktionen des Befehls geschieht folgendes:

1. **N**(ame) bezeichnet mit dem Namen FILE1.EXE eine Datei für einen späteren **L**(oad)- oder **W**(rite)-Befehl.
2. **N**(ame) bezeichnet mit dem Namen ebenfalls eine Datei, die als Variable eines später zu testenden Programmes zu laden ist.
3. **L**(oad) lädt die Datei FILE1.EXE in den Speicher.
4. **G**(o) beginnt die Programmabarbeitung, wie bei einem Aufruf von Betriebssystemebene aus.

Name

Ein weiteres Beispiel für die Kombination von Befehlen:

```
-NFILE1.EXE
-L
-NFILE2.DAT FILE3.DAT
-G
```

Jetzt können Sie mit FILE1.EXE die Datei bezeichnen, die mit dem Befehl L(oad) geladen werden soll. Danach werden mit dem Befehl **N**(ame) die Dateien FILE2.DAT und FILE3.DAT als Variable von FILE1.EXE bestimmt. Durch Eingabe von **G**(o) starten Sie das Programm. Es läuft so ab, als wäre auf Betriebssystemebene folgender Befehl eingegeben worden:

```
FILE1 FILE2.DAT FILE3.DAT
```

Beachten Sie folgendes:

Geben Sie jetzt den Befehl **W**(rite) ein, wird das zu testende Programm unter dem Namen FILE2.DAT gespeichert. Deshalb bezeichnen Sie vor jedem Befehl L(oad) oder **W**(rite) den Namen noch einmal genau.

Vier bestimmte Bereiche des Arbeitsspeichers beeinflußen Sie mit dem Befehl **N**(ame):

CS:5C FCB für Datei 1
CS:6C FCB für Datei 2
CS:80 Anzahl der Zeichen
CS:81 alle eingegebenen Zeichen

Ab der Adresse CS:5C wird für die erste mit **N**(ame) bezeichnete Datei ein FCB gebildet. Bezeichnen Sie einen zweiten Namen, so wird der FCB dafür ab CS:6C gebildet. Die Zahl der mit **N**(ame) eingegebenen Zeichen ist in CS:80 zu finden. Die Eingabe selbst wird ab CS:81 gespeichert.

Name

Ein typisches Beispiel für diesen Befehl ist:

```
C> DEBUG PROG.COM
-N PARAM1 PARAM2/C
-G
-
```

Das Programm läuft unter DEBUG so ab, als wäre es mit dem Befehl

```
C> PROG PARAM1 PARAM2/C
```

aufgerufen worden.

Output

Syntax

O Wert Byte

Zweck

gibt den Wert zum angegebenen Ein- / Ausgabekanal.

Bemerkungen

Geben Sie den Wert hexadezimal ein. Eine 16-Bit-Adresse ist erlaubt. Ein Beispiel:

```
02F8 4F
```

gibt den Wert 4Fh zum Ein- / Ausgabekanal 2F8h.

Proceed

Syntax

P [=[Adresse] [Wert]]

Zweck

führt das Programm schrittweise aus; behandelt dabei die Befehle LOOP und REP sowie Unterprogramme mit CALL und INT als ein Schritt

Bemerkungen

Bei Eingabe der Adresse wird ab dieser mit der Ausführung begonnen.

Enthält die Adresse eine Segmentangabe, dann werden das Segmentregister CS und der Programmzähler (Register IP) gesetzt. Fehlt das Segment, wird nur der Programmzähler gesetzt.

Mit der Eingabe von Wert, bestimmen Sie die Anzahl der auszuführenden Funktionen (maximal FFFFh). Dieser Befehl greift ebenso wie der Befehl T(race) auf den TRACE-Modus des Prozessors zurück und kann auch für das ROM benutzt werden. Er unterscheidet sich vom Befehl T(race) dadurch, daß er Unterprogramme, die mit dem Befehl CALL oder CALLF aufgerufen werden, komplett abarbeitet. Dies gilt auch für Programmschleifen mit dem Befehl LOOP oder Befehlsfolgen mit dem Präfix REP.

Folgende Befehlsfolge wird für das Beispiel benutzt:

```
3868:0100 B430      MOV     AH,30
3868:0102 CD21      INT     21
3868:0104 86E0      XCHG    AL,AH
3868:0106 3D0002    CMP     AX,0200
3868:0109 7309      JNB     0114
3868:010B BA102E    MOV     DX,2E10
3868:010E B409      MOV     AH,09
```

Geben Sie den Befehl ein:

```
P=100 4
```

erscheinen folgende Registeranzeigen:

```
AX=3000  BX=0000  CX=3955  DX=0000  SP=FFFE  BP=0000  SI=0000  DI=0000
DS=3868  ES=3868  SS=3868  CS=3868  IP=0102   NV UP EI PL NZ NA PO NC
3868:0102 CD21          INT   21

AX=0A03  BX=0500  CX=0000  DX=0000  SP=FFFE  BP=0000  SI=0000  DI=0000
DS=3868  ES=3868  SS=3868  CS=3868  IP=0104   NV UP EI PL NZ NA PO NC
3868:0104 86E0          XCHG   AL,AH

AX=030A  BX=0500  CX=0000  DX=0000  SP=FFFE  BP=0000  SI=0000  DI=0000
DS=3868  ES=3868  SS=3868  CS=3868  IP=0106   NV UP EI PL NZ NA PO NC
3868:0106 3D0002        CMP   AX,0200

AX=030A  BX=0500  CX=0000  DX=0000  SP=FFFE  BP=0000  SI=0000  DI=0000
DS=3868  ES=3868  SS=3868  CS=3868  IP=0109   NV UP EI PL NZ NA PO NC
3868:0109 7309          JNB   0114
_
```

Dabei wurde der Interrupt 21 an der Adresse 102 als ein Schritt behandelt.

Beachten Sie für die Befehle **P(roceed)** und **T(race)** folgende Besonderheit des Prozessors:

Wird ein Segmentregister mit einem Wert geladen, führt der Prozessor automatisch den folgenden Befehl mit aus. Diese Befehlsfolge kann nicht unterbrochen werden. Daher wird ein Einzelschritt nicht angezeigt. Benutzen Sie den Befehl **P(roceed)** für die Befehlsfolge:

```
3868:0101 1F        POP   DS
3868:0102 E921FE    CALL  1023
3868:0104 86E0      MOV   BX,AX
3868:0106 3D0002    CMP   AX,0200
```

Proceed

Geben Sie ein:

```
P=101
```

Das Programm DEBUG führt den Prozessorbefehl POP DS aus. Anschließend wird automatisch das Unterprogramm aufgerufen (CALL 1023). Erst vor dem Befehl MOV BX,AX stoppt die Abarbeitung des Programmes, und Sie können die aktuellen Register sehen. Soll das Programm vor Aufruf der Routine angehalten werden, muß an die erste Adresse des Prozessorbefehles CALL mit Hilfe der Anweisung **G**(o) ein Haltepunkt gesetzt werden.

Dies gilt ebenfalls für den DEBUG-Befehl **T**(race).

Quit

Syntax

Q

Zweck

beendet das Programm

Bemerkungen

Zu dem Befehl Q dürfen Sie keine weiteren Eingaben vornehmen. Das Programm DEBUG wird verlassen, ohne die bearbeitete Datei zu speichern. Sie kehren wieder zum Betriebssystem zurück.

Register

Syntax

R [Register]

Zweck

zeigt den Inhalt eines oder mehrerer Register der CPU.

Bemerkungen

Bezeichnen Sie keinen Registernamen, werden sämtliche Register mit ihrem momentanen Inhalt angezeigt. Zusätzlich wird die nächste Funktion angezeigt, auf die der Programmzähler zeigt.

Bei Eingabe des einzelnen Registernamens wird der Inhalt dieses Registers angezeigt, und Sie haben die Möglichkeit, den Inhalt zu verändern. Bei Eingabe von **RETURN** verändert sich der Wert nicht.

Tabelle 16.3 Register der Prozessoren 8086/8088

REGISTER	FUNKTION
AX	Akkumulator
BX	Adressregister
CX	Zählregister
DX	Adressregister für Ein- / Ausgabe
CS	Codesegment
DS	Datensegment
SS	Stapelsegment
ES	Extrasegment
SP	Stapelzeiger
BP	Basiszeiger
SI	Indexregister
DI	Indexregister
IP	Programmzähler
F	Statusregister

Wird ein anderer Registername als in der o. a. Tabelle aufgeführt eingegeben, erscheint die Fehlermeldung:

BR (Bad Register) Falsches Register

Tabelle 16.3 Kennzeichen für das F-Register

KENNZEICHEN	DEFINIERT		GELÖSCHT	
Überlauf	OV	Überlauf	NV	kein Überlauf
Richtung	DN	Vermindern	UP	Erhöhen
Unterbrechung	EI	Aktiviert	DI	nicht aktiviert
Vorzeichen	NG	Negative	PL	Plus
Null	ZR	Null	NZ	nicht null
zusätzlicher Übertrag	AC	Hilfsübertrag	NA	kein Hilfsübertrag
Parität	PE	Gerade	PO	Ungerade
Übertrag	CY	Übertrag	NC	kein Übertrag

Bei der Eingabe von **F** als Registername werden alle Kennzeichen ausgegeben (s.a. Tabelle 15.4). Die Kennzeichen werden in einer Zeile dargestellt, mit einem Bindestrich (-) als letztes Zeichen der Zeile. Danach ist der Cursor positioniert, und Sie können die gewünschten Änderungen eingeben. Das kann in beliebiger Reihenfolge geschehen. Die Eingabe wird mit **RETURN** abgeschlossen.

Geben Sie mehr als einen Wert für ein Kennzeichen ein, erscheint die Fehlermeldung

DF (Double Flag) Doppeltes Kennzeichen

und der alte Wert ändert sich nicht. Ebenso erhalten Sie eine Fehlermeldung, wenn Sie ein ungültiges Kennzeichen eingeben. Es erscheint die Meldung:

BF (Bad Flag) Falsches Kennzeichen

Geben Sie mehrere Kennzeichen in einer Zeile ein, werden Änderungen nur bis zum falschen Kennzeichen vorgenommen.

Register

Zu Beginn sind alle Segmentregister mit dem niedrigsten freien Speicherplatz belegt, der IP zeigt auf 100h und der SP auf FFEEh. Sämtliche Kennzeichen sind nicht gesetzt und die restlichen Register werden auf Null gesetzt. Geben Sie ein:

R

Es könnte folgender Prozessorstatus angezeigt werden:

```
AX=0000 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS =16A8 ES =16A8 SS =16A8 CS =16A8 IP =0100   NV UP DI PL NZ NA PO NC
16A8:0100 9F            LAHF
```

Geben Sie nur ein

RF

wird das Statusregister alleine angezeigt:

```
NV UP DI PL NZ NA PO NC
```

Jetzt können Sie andere Kennzeichen in beliebiger Reihenfolge eingeben, z. B.:

```
NV UP DI PL NZ NA PO NC - PL EI CY
```

Änderungen werden nicht direkt angezeigt. Durch erneute Eingabe des Registernamens erfolgt die Anzeige:

```
NV UP EI PL NZ AC PE CY -
```

Search

Syntax

S Bereich Liste
S Bereich Zeichenkette

Zweck

durchsucht den Bereich des Arbeitsspeichers nach den eingegebenen Werten und zeigt die Adressen der Werte.

Bemerkungen

Es dürfen ein oder mehrere Werte in die Liste eingegeben werden, die durch ein Komma oder eine Leertaste getrennt werden. Geben Sie eine Zeichenkette ein, muß diese von Anführungsstrichen eingeschlossen werden (s. a. Seite 16.7). Angezeigt wird jeweils die erste Adresse der Liste oder Zeichenkette. Ein Beispiel:

S CS:100 L10 41

Der Bereich wird nach dem Wert 41h abgesucht. Es könnten folgende Adressen ausgegeben werden:

04BA:0104
04BA:010D

Ist der eingegebene Wert nicht gefunden worden, erfolgt keine weitere Anzeige. Geben Sie eine Zeichenkette folgendermaßen ein:

S CS:0 L1000 "Zenith PC"

Trace

Syntax

T [=[Adresse] [Wert]]

Zweck

führt das Programm *schrittweise* aus, zeigt nach jedem Befehl alle Register sowie die nächste auszuführende Funktion.

Bemerkungen

Bei Eingabe der *Adresse* wird ab dieser mit der Ausführung begonnen.

Enthält die Adresse eine Segmentangabe, dann werden das Segmentregister CS und der Programmzähler (Register IP) gesetzt. Fehlt das Segment, wird nur der IP gesetzt.

Mit der Eingabe von *Wert* bestimmen Sie die Anzahl der auszuführenden Funktionen (maximal FFFFh). Dieser Befehl greift auf den TRACE-Modus des Prozessors zurück und kann auch für das ROM benutzt werden. Nach Eingabe von:

```
T
```

wird dann z. B. folgendes angezeigt:

```
AX=0000 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=16A8 ES=16A8 SS=16A8 CS=16A8 IP=0102   NV UP DI PL NZ NA PO NC
16A8:0102 0000          ADD [BX+SI],AL
```

Geben Sie ein

```
T=011A 10
```

werden mit der relativen Adresse 011Ah beginnend 16 (10h) Funktionen ausgeführt. Nach jeder Funktion werden alle Register angezeigt. Dies geschieht ohne Pause. Sie können deshalb mit der Tastenkombination **CTRL-S** die Bildschirmausgabe jederzeit unterbrechen, um die Registerinhalte zu studieren.

Der Befehl T(race) führt im Gegensatz zum Befehl P(roceed) jeden Processorbefehl einzeln aus und zeigt danach immer die Registerinhalte. Auch für diesen Befehl gilt, daß nach Laden eines Segmentregisters automatisch der folgende Befehl ausgeführt wird, bevor DEBUG die Kontrolle über den Prozessor zurückbekommt.

Unassemble

Syntax

U [Adresse] [L Wert]
U [Bereich]

Zweck

wandelt Speicherwerte in die entsprechenden *Maschinenbefehle* um und zeigt diese mit Adresse und Wert an.

Bemerkungen

Die Ausgabe sieht aus wie das Listing eines Assemblerprogrammes.

Geben Sie den Befehl ohne Zusatz ein, werden 20 Hexadezimalwerte disassembliert und angezeigt. Geben Sie einen Bereich an, werden alle Werte dieses *Bereiches* angezeigt.

Soll die Ausgabe ab einer bestimmten Adresse beginnen, geben Sie diese Adresse ein.

Wird für *Bereich* ein Wert eingegeben, werden die entsprechenden Bytes disassembliert und angezeigt.

Geben Sie ein:

```
U 17BB:0C00
```

wird z. B. folgendes angezeigt:

```
17BB:0C00  E84CFE     CALL  0A4F
17BB:0C03  7306       JNB   0C0B
17BB:0C05  B80200     MOV   AX,0002
17BB:0C08  E9DF01     JMP   0DEA
17BB:0C0B  33D2       XOR   DX,DX
17BB:0C0D  33C9       XOR   CX,CX
17BB:0C0F  8B1ECF2C   MOV   BX,[2CCF]
17BB:0C13  B002       MOV   AL,02
17BB:0C15  B442       MOV   AH,42
17BB:0C17  CD21       INT   21
17BB:0C19  E81CFF     CALL  0B38
17BB:0C1C  7503       JNZ   0C21
17BB:0C1E  2D0002     SUB   AX,0200
```

Geben Sie dagegen ein

```
U 17BB:0C00 0C08
```

werden nur die Bytes zwischen den Speicheradressen 17BB:0C00 und 17BB:0C08 disassembliert:

```
17BB:0C00  E84CFE     CALL  0A4F
17BB:0C03  7306       JNB   0C0B
17BB:0C05  B80200     MOV   AX,0002
17BB:0C08  E9DF01     JMP   0DEA
```

Ebenso ist es möglich, für die Segmentadresse die Bezeichnung des Segmentregisters einzugeben. Damit wird derselbe Bereich des Speichers bezeichnet.

Ändern Sie während der Testphase Teile des Programmes, wird auch bei der Disassemblierung der neue Kode angezeigt.

Write

Syntax

W [Adresse [Laufwerk Sektor Anzahl]]

Zweck

schreibt die getestete Datei auf Diskette oder Winchester

Bemerkungen

Die Eingabe des Befehles **W** ohne weitere Zusätze schreibt die Datei, beginnend ab der Adresse CS:100, mit der Anzahl der Bytes, die in den Registern BX:CX definiert sind, auf die Platte.

Benennen Sie eine Adresse, wird ab dieser Adresse geschrieben.

Wurden die Befehle **G**(o), **P**(roceed) oder **T**(race) vorher benutzt, müssen erst die Register BX:CX neu definiert werden. Danach kann der Befehl **W** ohne Zusatz eingegeben werden.

Ändern Sie eine getestete Datei, wird sie nur korrekt auf die Platte zurückgeschrieben, solange Sie nicht die Größe der Datei verändern bzw. die Register BX:CX neu definieren. Vor Aufruf des Befehles **W**(rite) müssen Sie den Dateinamen definieren. Das geschieht entweder beim Aufruf der Datei mit dem Programm DEBUG zusammen oder durch den Befehl **N**(ame). In beiden Fällen wird ab CS:5C der Dateiname gespeichert.

Der Befehl **W**(rite) schreibt die Datei immer an den Platz auf der Platte, von wo die Datei geladen wurde. Dabei werden dann die vorherigen Informationen überschrieben. Während dieses Vorganges wird die Anzahl der zu schreibenden Bytes angezeigt.

Definieren Sie sämtliche Variable bei der Befehlseingabe, wird der Speicherinhalt direkt in einen absoluten Sektor der Platte geschrieben. Da Sie das Betriebssystem dabei umgehen, ist dieses **E X T R E M** gefährlich. Die Laufwerke werden dabei wie folgt bezeichnet:

0=A:, 1=B:, 2=C:, 3=D:, 4=E:, 5=F:, 6=G:, 7=H:.

Ein Beispiel:

```
W CS:100 1 37 2B
```

Sie schreiben jetzt den Speicherinhalt ab der Adresse CS:100 auf das Laufwerk B: (1). Dabei werden 43 Sektoren (2Bh) beschrieben, beginnend ab dem absoluten Sektor 55 (37h).

Der Befehl W(rite) fragt zuerst den Namenszusatz der zu speichernden Datei ab. Daher können keine Änderungen in Dateien mit dem Zusatz .EXE oder .HEX gespeichert werden. Laden Sie diese Dateien mit einem anderen Namenszusatz oder ändern Sie den Namen vorher mit dem DEBUG-Befehl N(ame).

Speichern Sie Änderungen für Dateien mit dem Zusatz .COM, ignoriert der Befehl W(rite) immer die eingegebene Adresse. Automatisch wird beginnend ab der Adresse DS:100 gespeichert.

Fehlermeldungen

Bei einer Falscheingabe erscheint eine der folgenden Fehlermeldungen. Dadurch wird der eingegebene Befehl nicht ausgeführt. Das Programm DEBUG wird dadurch nicht abgebrochen. Geben Sie den Befehl anschließend richtig ein, können Sie mit der Bearbeitung der Datei fortfahren.

BF (Bad Flag) Falsches Kennzeichen

ERKLÄRUNG:
Es wurde versucht, ein Kennzeichen des Statusregisters mit einer falschen Buchstabenkombination zu ändern. Gültige Werte sind in der Tabelle 16.4 aufgelistet.

BP (Too many Breakpoints) Zu viele Stopadressen

ERKLÄRUNG:
Es wurden mehr als die erlaubten 10 Stopadressen mit dem Befehl G(o) eingegeben. Geben Sie den Befehl erneut ein.

BR (Bad Register) Falsches Register

ERKLÄRUNG:
Der Befehl R(egister) wurde mit einem falschen Registernamen eingegeben. Gültige Namen sind in der Tabelle 16.3 aufgelistet.

DF (Double Flag) Doppeltes Kennzeichen

ERKLÄRUNG:
Es wurde versucht, mit einem Befehl zwei Werte für ein Kennzeichen einzugeben.

Access denied	Zugriff nicht möglich

ERKLÄRUNG:
Die fgerade bearbeitete Datei ist mit dem Programm ATTRIB als Nur-Lese-Datei markiert worden. Sie kann daher auf Platte nicht geändert werden. Entfernen Sie vorher mit dem Programm ATTRIB den Schreibschutz.

Error in .EXE or .HEX file	Fehler in einer .EXE- oder .HEX-Datei

ERKLÄRUNG:
Die Datei enthält ungültige Werte und kann nicht bearbeitet werden.

EXE and HEX files cannot be written	.EXE- und .HEX-Dateien können nicht beschrieben werden

ERKLÄRUNG:
Die gerade bearbeitete Datei hat den Namenszusatz EXE oder HEX. Diese Dateien werden beim Aufruf von DEBUG im Arbeitsspeicher verändert und können daher nicht wieder auf Platte gespeichert werden. Wollen Sie Änderungen speichern, laden Sie diese Datei als Binärdatei mit einem anderen Zusatz.

EXEC failure	Fehler durch Systemanruf

ERKLÄRUNG:
Diese Fehlermeldung kann durch eine der folgenden Ursachen hervorgerufen werden:

- Die aufgerufene Datei existiert nicht in der Form, in der sie aufgerufen wurde.
- Das aufgerufene .EXE-Programm enthält im Kopf nicht die von MS-DOS geforderten Informationen.
- Der Arbeitsspeicher ist zu klein.
- Es wurde ein falscher Funktionsaufruf benutzt. Funktion Nummer 0, 1 oder 3 ist richtig.

Fehlermeldungen

File creation error — Falsche Dateianlage

ERKLÄRUNG:
Im Inhaltsverzeichnis der Diskette oder des Winchesterlaufwerkteiles ist kein Platz mehr, um eine weitere Datei zu speichern. Diese Meldung erscheint auch, wenn der Dateiname schon für ein Inhaltsverzeichnis vergeben ist.

File not found — Datei nicht vorhanden

ERKLÄRUNG:
Die bezeichnete Datei ist nicht auf der Platte. Überprüfen Sie die Eingabe bzw. den mit dem Befehl N(ame) definierten Namen.

Incorrect Dos version — Falsche DOS-Version

ERKLÄRUNG:
Sie benutzen das Programm DEBUG unter einer DOS-Version, die nicht 2.0 oder höher ist.

Insufficient memory — Arbeitsspeicher zu klein

ERKLÄRUNG:
Die freie Kapazität des Arbeitsspeichers reicht nicht aus, um die bezeichnete Datei zu laden.

Insufficient disk space — Nicht genug Speicherkapazität auf der Platte

ERKLÄRUNG:
Die Speicherkapazität der Platte, auf die geschrieben werden soll, reicht nicht aus für die Datei. Wechseln Sie die Diskette (Teilbereich der Winchester).

Invalid drive specification — Falsche Laufwerkbezeichnung

ERKLÄRUNG:
Sie haben im Befehl L(oad) oder W(rite) das Laufwerk, auf das zugegriffen werden soll, falsch bezeichnet. Überprüfen Sie Ihre Eingabe.

Fehlermeldungen

Parity error or nonexistant memory error detected	Fehler des Arbeitsspeichers gefunden

ERKLÄRUNG:
Das Programm hat einen Fehler während der Überprüfung der Speicherchips gefunden. Starten Sie das System neu. Wird der Fehler wiederholt gemeldet, wenden Sie sich an Ihren Zenith-Händler.

Write protect error writing drive X	Schreibfehler, da die Diskette X schreibgeschützt ist

ERKLÄRUNG:
Sie haben versucht, mit dem Befehl W(rite) auf eine schreibgeschützte Diskette zu schreiben. Wechseln Sie die Diskette oder entfernen Sie den Schreibschutz.

Teil V

Dienstprogramme für Winchesterlaufwerke

Dienstprogramme für Winchesterlaufwerke

Dieser Teil erläutert ausführlich die vier Dienstprogramme für Winchesterlaufwerke, die auf der zweiten Systemdiskette mitgeliefert werden.

Benutzer des Zenith PC, die kein Winchesterlaufwerk besitzen, brauchen diese Kapitel nicht zu lesen. Ebenso nicht Benutzer, die Ihr Winchesterlaufwerk nicht neu aufteilen wollen.

Die Erklärungen erfolgen in allgemein verständlichen Worten, die auch der Benutzer verstehen kann, der noch keine Erfahrung mit Mikrocomputern hat. Es wird lediglich erwartet, daß der Leser die Begriffe kennt, die im ersten Teil dieses Handbuches erklärt wurden.

Die Reihenfolge der Befehlserklärungen erfolgt so, wie diese im Programm wahrscheinlich auch von Ihnen benutzt werden. Trotzdem kann es vorkommen, daß Sie die Programme in einer anderen Reihenfolge benutzen müssen.

Dieser Teil des Handbuches besteht aus den folgenden vier Kapiteln:

Kapitel 17: PREP erklärt die Befehlseingaben und den Ablauf des Programmes, das Ihre Winchester zur Datenaufnahme vorbereitet.

Kapitel 18: PART erklärt die Befehlseingaben und den Ablauf des Programmes, das ein Winchesterlaufwerk in maximal vier logische Laufwerke aufteilt.

Kapitel 19: ASGNPART erklärt die Befehlseingaben und den Ablauf des Programmes, das einen Teilbereich der Winchester als logisches Laufwerk mit einem Laufwerknamen bezeichnet.

Kapitel 20: SHIP erklärt die Befehlseingabe und den Ablauf des Programmes, das die Winchesterlaufwerke zum Transport vorbereitet.

Kapitel 21: DETECT erklärt die Befehlseingaben und den Ablauf des Programmes, das fehlerhafte Sektoren findet und für eine weitere Benutzung sperrt.

Für die Befehlseingabe dieser externen Befehle gelten die in Kapitel 5 erläuterten Vorschriften.

Kapitel 17

PREP

Übersicht

Dieses Programm magnetisiert Ihre Winchesterplatten und bereitet sie damit zur Datenaufnahme vor. Das Programm ist auf der zweiten Systemdiskette enthalten.

Das Programm werden Sie wahrscheinlich sehr selten benutzen (wenn überhaupt). Mit diesem Programm löschen Sie sämtliche Daten, die auf Ihrer Winchester gespeichert waren. Daher sollten Sie immer erst alle Dateien auf Disketten kopieren (s. a. BACKUP).

Das in Ihrem Zenith PC eingebaute Winchesterlaufwerk wurde bereits im Werk mit dem Programm PREP vorbereitet. Es ist nicht notwendig, daß Sie diesen Vorgang wiederholen. Sie sollten das Programm nur dann benutzen, wenn wiederholt Fehler beim Plattenzugriff gemeldet werden, und Sie diese Fehler auch nicht mit dem Programm DETECT beheben können.

Ist das Programm beendet, müssen Sie das System vom Diskettenlaufwerk aus neu starten (s. a. Kapitel 6).

Aufbau eines Winchesterlaufwerkes

Winchesterlaufwerke sind in unterschiedlichen Größen und verschiedenen Kapazitäten erhältlich. Sie haben alle gemeinsame Merkmale. Diese Laufwerke bestehen aus mehreren harten, nicht biegsamen Platten. Daher auch manchmal der Name *Hard-Disk*. Die einzelnen Platten sind gewöhnlich aus Aluminium, das mit einer dünnen gleichmäßigen Schicht Eisenoxid oder Kobalt überzogen ist. Im Gegensatz dazu besteht eine Diskette aus einer biegsamen Plastikscheibe, die ebenfalls mit einer dünnen Schicht Eisenoxid überzogen ist. Daher nennt man eine Diskette auch *Floppy-Disk*.

Platten

Die *Platten* eines Winchesterlaufwerkes befinden sich in einem versiegelten, staubfreien Gehäuse. Da der Schreib-Lese-Kopf in sehr geringem Abstand über der Platte angebracht ist, würde ein Staubkorn den Kopf mit der Platte in Berührung bringen (Head-Crash).

Es gibt Winchesterlaufwerke, die fest in das Gehäuse eingebaut werden (*Festplatte*), und Laufwerke, die ausgetauscht werden können (*Wechselplatte*).

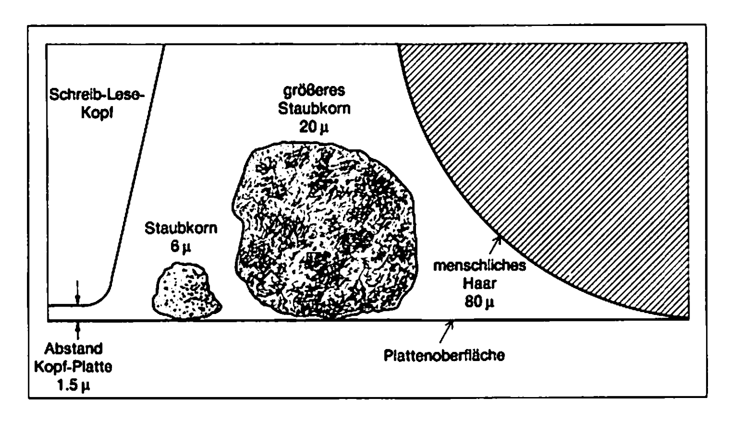


Bild 17.1
Größenverhältnisse

Schreib-Lese-Kopf

Der *Schreib-Lese-Kopf* ist ein Elektromagnet, der vorwärts und rückwärts über die Platten bewegt wird. Die Platten bewegen sich dabei mit ca. 3000 U/Min. Der Abstand zwischen Magnet und Plattenoberfläche ist dabei sehr gering. Die Bewegungen des Magneten über die Platten werden auch „Stepping" genannt. Die Maßeinheit zum Erfassen dieser Bewegung ist die „STEP-RATE".

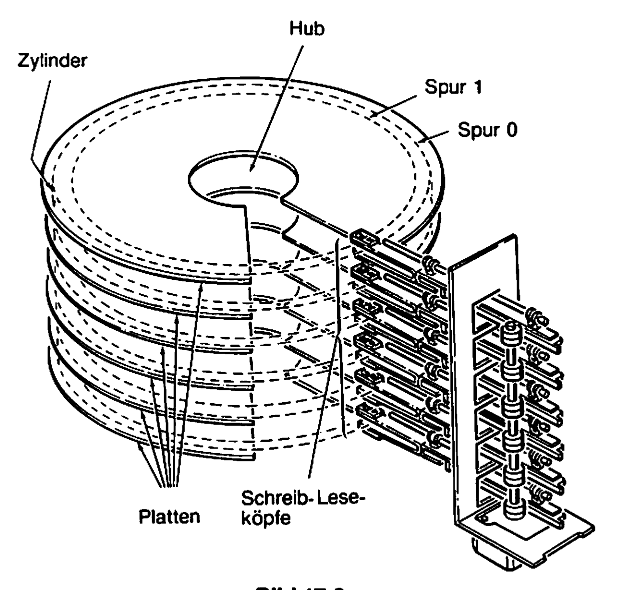


Bild 17.2
Typischer Aufbau eines Winchesterlaufwerkes

Aufbau eines Winchesterlaufwerkes

Aufteilung in logische Einzellaufwerke

Wegen der enormen Speicherkapazität eines Winchesterlaufwerkes ist es wünschenswert, dieses in kleinere Bereiche aufzuteilen. Die einzelnen Teile des gesamten Laufwerkes werden *Teilbereiche genannt. Die Bereiche werden wie separate Laufwerke behandelt. Es ist möglich, das Winchesterlaufwerk maximal in vier Einzellaufwerke (Teilbereiche oder Partitions) aufzuteilen. Damit erreichen Sie Ihre auf der Winchester gespeicherten Daten schneller.*

Sektoren

Die auf einem Laufwerk gespeicherten Daten werden in *Sektoren* untergebracht. Der Sektor ist der kleinste Bereiche, in dem Daten gespeichert werden können. Das Betriebssystem MS-DOS arbeitet mit einer Sektorgröße von 512 Bytes. Disketten- und Winchesterlaufwerke besitzen die gleiche Sektorgröße.

Spuren und Zylinder

Die Oberfläche der Platten ist in einzelne konzentrische Ringe aufgeteilt, die *Spuren* (Tracks) genannt werden. Diese Spuren sind mit den Spuren einer Diskette vergleichbar. Mit dem Programm PREP wird jede Spur in 17 *Bereiche (Sektoren)* unterteilt. Die Spuren, die auf allen Platten gleich weit von der Außenkante entfernt sind, werden zu *Zylindern* zusammengefaßt. Die Schreib-Lese-Köpfe können sämtliche Daten in einem Zylinder erreichen, ohne bewegt zu werden.

Das Winchesterlaufwerk Ihres Z-200 mit 20 Mb Speicherkapazität hat vier Schreib-Lese-Köpfe. Diese können pro Zylinder auf 68 Bereiche zugreifen (4 Spuren à 17 Bereiche), ohne daß sie bewegt werden müssen. Das ergibt eine Speicherkapazität pro Zylinder von 34.816 Bytes (34 Kb). Bei einer Gesamtzahl von 615 Zylindern hat die Winchester somit eine Gesamtspeicherkapazität von 21.411.840 Bytes (20 Mb). Teilen Sie Ihr Laufwerk in vier gleiche Teile, ergibt das eine Nettokapazität je logischem Laufwerk von 5.352.960 Bytes.

Programmaufruf

Syntax

[d:] [Weg] PREP

dabei bedeutet:

d:	vor dem Programmnamen bestimmt das aufzurufende Laufwerk
Weg	vor dem Programmnamen bestimmt das aufzurufende Verzeichnis

Bemerkungen

Das Dienstprogramm PREP führt folgende Schritte aus:

1. Initialisierung der Plattenoberflächen

2. Test des Speicherverhaltens der Platten

3. Sperrung fehlerhafter Sektoren

Damit sind alle Daten der Platte gelöscht. Sie sollten daher unbedingt vor Aufruf des Programmes alle Dateien mit dem Programm BACKUP auf Diskette sichern.

Programmablauf

Rufen Sie das Programm mit seinem Namen auf. Geben Sie ein:

PREP

Nach Aufruf des Programmes erscheint folgender Text:

```
                PREP version 3.01
Copyright (C) 1985, Zenith Data Systems Corporation

The PREP utility helps you to:

   * Initialize surface of Winchester disk
   * Test data retention capabilities of Winchester disk media
   * Isolate questionable disk sectors

PREP will prompt you for the winchester drive unit number.
Then PREP displays messages as it operates on the disk.

Caution: Using PREP will destroy all files on your Winchester disk. Winchester disks
supplied by Zenith are normally prepared by PREP before they are shipped. Users of
these disks will need to use PREP only, if they are consistently encountering an
unreasonable number of disk access errors. Do not use PREP until you have transferred
your Winchester disk files to floppy disks.

Do you wish to proceed with PREP (Y/N)_
```

Es wird hier noch einmal darauf hingewiesen, daß Sie dieses Programm erst aufrufen sollten, wenn Sie sämtliche Dateien der Winchester auf Diskette gesichert haben. Wird die Platte neu initialisiert, sind anschließend alle vorhandenen Daten gelöscht.

In der letzten Zeile werden Sie gefragt, ob Sie mit dem Programm fortfahren möchten. Geben Sie jetzt ein **N**(ein) ein, verlassen Sie das Programm und es erscheint wieder das Prompt des Betriebssystemes. Geben Sie ein **Y** für Ja ein, erscheint folgende Aufforderung:

Winchester drive unit number (0-3):	Nummer des Winchesterlaufwerkes (0-3):

Nachdem Sie die Nummer angegeben haben, erscheint folgende Aufforderung:

Type P to proceed	Zum Fortfahren betätigen Sie P

Jetzt wird das Programm nur noch nach Eingabe von P fortgesetzt. Jede andere Eingabe beendet das Programm. Geben Sie ein P ein, wird mit der Initialisierung begonnen.

Initialisierung der Plattenoberflächen

Nachdem Sie mit Eingabe von P das Programm gestartet haben, erscheint die Meldung:

Initializing the disk...	Die Platten werden initialisiert

Jetzt werden, ähnlich wie mit dem Befehl FORMAT für die Diskette, die einzelnen Sektoren angelegt. Damit wird die Platte für den folgenden Test vorbereitet. Ist der Vorgang beendet, erscheint die Meldung:

Initializing the disk completed	Die Platten werden initialisiert beendet

Programmablauf

Test der Platte

Mit diesem Test wird das Speichervermögen der Platte untersucht. Der Test besteht aus sechs Durchläufen. In jedem Lauf werden bestimmte Daten in jeden Sektor geschrieben und wieder gelesen. Damit wird überprüft, ob die in jedem Sektor gespeicherten Daten einwandfrei gelesen werden können. Der jeweilige Lauf wird am Bildschirm angezeigt:

Media test in progress, pass n	Test des Speichervermögens Lauf n

Dabei ist *n* die Nummer des momentanen Laufes.

VORSICHT: Für die Winchester mit 20 Mb Speicherkapazität dauert dieser Test ca. 3 Stunden.

Jeder Testlauf wird mit einem anderen Wert durchgeführt. Findet das Programm dabei fehlerhafte Sektoren, werden sie gespeichert und anschließend in eine Tabelle eingetragen.

Initialisierung des geschützten Bereichs

Nach Beendigung des Testes beginnt das Programm mit dem Einrichten eines geschützten Bereiches auf der Platte. Dieser Bereich wird in den ersten Sektoren angelegt. Es erscheint wieder die Meldung:

Initializing the disk...	Die Platte wird initialisiert...

Erscheint das Wort completed in der Zeile, ist auch dieser Vorgang beendet. Es erscheint nun wieder das Prompt des Betriebssystems. Auf die Platte können Sie erst zugreifen, nachdem Sie von der Diskette aus neu gebootet haben.

Der geschützte Bereich

Diese Informationen über das Dienstprogramm sind nicht erforderlich, um das Programm aufzurufen oder es ablaufen zu lassen. Diese Informationen werden für den interessierten Leser weitergegeben, der mehr über die Arbeitsweise wissen möchte.

Das Programm PREP schreibt verschiedene Informationen zur Verwaltung der Platte auf den ersten Sektor des ersten Zylinders. Auf dem vorletzten und letzten Zylinder werden ebenfalls Informationen zur Plattenverwaltung gespeichert. Diese Bereiche werden unter dem Begriff *geschützter Bereich* zusammengefaßt. Sie werden während der Initialisierung dieses Bereiches auf die Platte geschrieben.

Die wichtigsten Informationen sind der *Boot-Satz*, die *Partition-Tabelle* und die *Tabelle der fehlerhaften Sektoren.* Der Boot-Satz und die Partition-Tabelle werden in den ersten Sektor (Kopf 0, Zylinder 0, Sektor 1) geschrieben. Die Tabelle der fehlerhaften Sektoren wird in den vorletzten und letzten Zylinder geschrieben.

Diese Informationen werden benötigt, um von der Winchester zu booten. Nur so kann von einem Teil der Winchester (logisches Laufwerk) gebootet werden. Ebenso werden die Informationen benötigt, um fehlerhafte Sektoren für den weiteren Gebrauch zu sperren. Damit wird erreicht, daß sie erst gar nicht formatiert werden.

Der Boot-Satz

Der *Boot-Satz* ist ein kleines Programm zur Winchesterverwaltung. Es enthält alle Informationen, die benötigt werden, um von der Winchester zu booten.

Dieses Programm wird von PREP während der Initialisierung des geschützten Bereiches in den ersten Sektor geschrieben.

Geben Sie den Befehl ein, von der Winchester zu booten, wird der Boot-Satz von der Platte in den Arbeitsspeicher geladen. Er unterscheidet sich vom MS-DOS BOOT-LOADER. Der Boot-Loader ist ein Programm, um das Betriebssystem zu laden. Er wird in jeden Teil der Platte geschrieben, der Boot-Satz nur in den ersten Sektor.



Der geschützte Bereich

Das Boot-Byte

Ist der Boot-Satz im Arbeitsspeicher, greift er auf einen Teil der Platte zu. Der Teil, auf den zugegriffen werden soll, wird durch das *Boot-Byte* bestimmt.

Der Boot-Satz lädt vom bezeichneten Teil den ersten Sektor in den Speicher. Der Sektor wird nach einem Zeichen abgesucht, das das Programm Boot-Loader kennzeichnet. Wird das Zeichen gefunden, wird dem Boot-Loader die weitere Kontrolle übergeben.

Tabelle der logischen Laufwerke

Diese Tabelle enthält alle Informationen über die bezeichneten Laufwerke, in die die Platte aufgeteilt werden kann. Sie enthält die folgenden Informationen:

- Boot-Byte
- Startzylinder, Kopf & Sektor
- Endzylinder, Kopf & Sektor
- Betriebssystem
- Anzahl der dem Teil vorausgehenden Sektoren
- Anzahl der vom Teil belegten Sektoren

Die Tabelle wird am Ende des Boot-Satzes gespeichert. Jeder Eintrag beträgt 16 Bytes. Auf der folgenden Seite finden Sie eine Übersicht, die im Kapitel 18 ausführlich erklärt wird.

Das Boot-Byte zeigt dem Boot-Satz an, von welchem Teil des Laufwerkes aus das System gestartet werden kann. Das Dienstprogramm PART beschreibt dieses Boot-Byte. Es wird zu dem Wert 80h die Nummer des Winchesterlaufwerkes (0 - 3) addiert. Dieser Wert wird dann in das Boot-Byte des gewünschten Teiles geschrieben. Gleichzeitig werden alle anderen Bytes auf 00 gesetzt. Damit erkennt der Boot-Satz, daß bei einem Wert zwischen 80h und 83h dieser Teil der Platte ein bootfähiges System enthält. Er lädt von diesem Teil den BOOT-LOADER des Betriebssystems und übergibt ihm die weitere Kontrolle.

Struktur der Tabelle

Die Tabelle kann Informationen für maximal vier logische Laufwerke speichern. Die Tabelle 17.1 zeigt die Struktur des 16-Byte-Feldes. Die Startadresse (innerhalb des Boot-Satzes) für jedes Laufwerk ist:

- 01BE für Teilbereich #1
- 01CE für Teilbereich #2
- 01DE für Teilbereich #3
- 01EE für Teilbereich #4

Tabelle 17.1 Struktur des Eintrages

OFFSET	BYTES	NAME	ERKLÄRUNG
00	1	Boot	Boot-Byte
01	1	Start	erster Kopf des Bereiches
02	1		erster Sektor des Bereiches
03	1		erster Zylinder des Bereiches
04	1	Sys	System-Byte
05	1	Ende	letzter Kopf des Bereiches
06	1		letzter Sektor des Bereiches
07	1		letzter Zylinder des Bereiches
08	4	rel. sec.	Anzahl der dem Teilbereich vorausgehenden Sektoren
0C	4	num.sec.	Größe des Teilbereiches in Sektoren

Der geschützte Bereich

Tabelle der fehlerhaften Sektoren

Diese Tabelle enthält Informationen über beschädigte Sektoren, auf die nicht zugegriffen werden kann. Diese Tabelle verhindert, daß von MS-DOS darauf geschrieben wird und es zu Datenverlusten kommt. Die Tabelle wird im vorletzten und letzten Sektor der Platte gespeichert. Es können maximal die Adressen von 168 Sektoren eingetragen werden. Jeder Eintrag enthält die Nummer des Kopfes, des Zylinders und des Sektors. Ist noch kein Eintrag vorhanden, werden alle Bytes auf 00h gesetzt.

Tabelle 17.2 Einträge der fehlerhaften Sektoren

BYTES	FEHLERHAFTER SEKTOR
3	Kopf, Zylinder, Sektor des 1. fehlerhaften Sektors, der von PREP gefunden wurde
3	Kopf, Zylinder, Sektor des 2. fehlerhaften Sektors, der von PREP gefunden wurde
3	Kopf, Zylinder, Sektor des 3. fehlerhaften Sektors, der von PREP gefunden wurde
3	Kopf, Zylinder, Sektor des 4. fehlerhaften Sektors, der von PREP gefunden wurde
. . .	. . .
3	Kopf, Zylinder, Sektor des 168. fehlerhaften Sektors, der von PREP gefunden wurde
3	Letzter Eintrag der Tabelle (immer 000)
2	Prüfsumme
2	Reservierte Bytes
1	Erkennungsbyte (Platte formatiert)

512	Gesamtzahl der Bytes in jeder Tabelle

Der geschützte Bereich

Während des Testvorganges speichert das Programm PREP alle fehlerhaften Sektoren, die es entdeckt hat. Nach der Initialisierung des geschützten Bereiches trägt es diese Sektoren in die Tabelle ein. Zur Sicherheit wird von der Tabelle noch eine Kopie angelegt.

Die erste Tabelle, die Tabelle A, wird ständig benutzt. Sollte sie jedoch selbst beschädigt sein, wird die zweite Tabelle, die Tabelle B benutzt. Tabelle B ist eine Kopie der Tabelle A und wird zusätzlich gespeichert.

Für jede Tabelle wird eine eigene Prüfsumme errechnet und in der jeweiligen Tabelle mit abgespeichert.

Initialisieren Sie mit dem Programm PREP die Winchester noch einmal, vergleicht das Programm die alte und die neue Prüfsumme, um festzustellen, ob eine Änderung eingetreten ist.

Kann die Tabelle A nicht gelesen werden oder differieren beide Prüfsummen, wird die Tabelle B gelesen. Kann die Tabelle B nicht gelesen werden, dann sind keine Informationen mehr über beschädigte Sektoren erhältlich. Bei einer neuen Formatierung der Platte werden dann alle Sektoren formatiert.

Überprüfen Sie jetzt mit dem Dienstprogramm DETECT die Platte, ist keine Tabelle mehr vorhanden, in die fehlerhafte Sektoren eintragen werden können. Das Programm DETECT überprüft dann die gesamte Platte und legt eine neue Tabelle an.

Fehlermeldungen

Bad sector count exceeded for this drive — Anzahl der fehlerhaften Sektoren ist zu groß für diese Platte

ERKLÄRUNG:
Es wurden mehr als 168 fehlerhafte Sektoren festgestellt. Dies könnte ein Hardwarefehler sein. Starten Sie das Programm PREP neu. Erscheint die Fehlermeldung erneut, wenden Sie sich an Ihren Händler.

Error during formatting of the drive — Fehler während des Initialisierungsvorganges

ERKLÄRUNG:
Die Winchesterkontrollerkarte könnte defekt sein. Setzen Sie sich mit Ihrem Händler in Verbindung.

Boot sector is bad — Bootsektor ist defekt

ERKLÄRUNG:
Der Bootsektor kann nicht richtig initialisiert werden. Starten Sie das Programm erneut. Erscheint der Fehler wieder, kann die Platte nicht benutzt werden. Wenden Sie sich an Ihren Händler.

Unable to communicate with the Winchester controller — Winchesterkontroller nicht vorhanden

ERKLÄRUNG:
Das Programm findet den Winchesterkontroller nicht. Es ist möglich, daß die Karte nicht richtig installiert ist. Überprüfen Sie alle Anschlüsse und starten Sie das Programm dann neu. Erscheint der Fehler wieder, ist die Karte wahrscheinlich defekt.

PART

Übersicht

Dieses Programm teilt ein Winchesterlaufwerk Ihres Zenith PC in verschiedene einzelne logische Laufwerke auf. Die logischen Laufwerke (maximal 4) können dann separat aufgerufen werden. Ebenso können Sie mit diesem Programm ein Teillaufwerk der Winchester als Startlaufwerk festlegen. Das Programm ist auf der zweiten Systemdiskette enthalten.

Das Programm führt folgende Schritte aus:

- Bestimmung eines Startlaufwerkes
- Aufteilung des Speicherplatzes der Platte in MS-DOS Bereiche durch Festlegung des Start- und Schlußzylinders.
- Löschung eines MS-DOS zugeteilten Bereichs.

Das in Ihrem Zenith PC eingebaute Winchesterlaufwerk wurde bereits im Werk mit dem Programm PART vorbereitet. Es ist ein Bereich für MS-DOS zugeteilt worden.

Ändern Sie die Bereiche auf der Platte, können Sie damit die bereits gespeicherten Daten *zerstören*. Sichern Sie daher immer vor Aufruf des Programmes die auf der Platte befindlichen Dateien mit dem Programm BACKUP.

Programmaufruf

Syntax

[d:] [Weg] PART

dabei bedeutet:

d:	vor dem Programmnamen bestimmt das aufzurufende Laufwerk
Weg	vor dem Programmnamen bestimmt das aufzurufende Verzeichnis

Bemerkungen

Die Speicherkapazität eines Winchesterlaufwerkes ist sehr viel größer als die einer Diskette. Sie errechnet sich aus der Gesamtkapazität abzüglich der durch das Programm PREP gesperrten Sektoren. Daher kann es im Laufe der Zeit sehr unübersichtlich werden, mit der Platte zu arbeiten. Außerdem braucht das Betriebssystem länger, um eine einzelne Datei wiederzufinden. Deshalb haben Sie die Möglichkeit, die Kapazität der Platte aufzuteilen. Damit Sie auf die Einzelteile leicht und schnell zugreifen können, werden diese als einzelne Laufwerke bestimmt. Mit dem Programm PART kann ein Winchesterlaufwerk in bis zu vier Einzellaufwerke unterteilt werden. Einschließlich des Diskettenlaufwerkes arbeiten Sie dann mit 5 Laufwerken, die mit den Buchstaben A bis F bezeichnet werden.

Das Programm PART ist menügesteuert. Ihre Eingaben können Sie jedesmal aus den angezeigten Möglichkeiten auswählen. Es wird in diesem Kapitel jede Möglichkeit ausführlich beschrieben. Das am Bildschirm erscheinende Menü ist abgedruckt. Haben Sie eine Eingabe vorgenommen, erscheint sofort ein neues Menü, das die Eingabe berücksichtigt.

Programmaufruf

Nach Aufruf des Programmes erscheint folgender Text auf dem Bildschirm:

PART version 3.01
Copyright (C) 1985, Zenith Data Systems Corporation

The PART utility helps you to change the arrangement of the DOS partitions on your Winchester disk. Part displays a table showing the types of each partitions, the start cylinder, the end cylinder and the amount of disk space allocated to each partition (in percentages and in kilobytes)

CAUTION: Using PART can destroy all DOS files on your Winchester disk.

Do not use Part until you have transferred copies of your Winchester disk files to floppy disks.

Do you wish to continue (Y/N)?_

Die bereits beschriebene Warnung wird hier noch einmal wiederholt. Beantworten Sie die Frage in der letzten Zeile, ob Sie fortfahren möchten, mit einem **N**(ein), verlassen Sie sofort das Programm und sind wieder im Betriebssystem. Geben Sie hier ein **Y** für Ja ein, erscheint folgendes Bild:

F1 - Partition Winchester drive 1

E - Exit to DOS

Enter selection <F1 or E >

Es ist möglich, an Ihren Z-200 PC bis zu drei Winchesterlaufwerke anzuschließen. Obwohl in den meisten Fällen dies nicht voll ausgenutzt wird, können Sie mit diesem Programm alle drei Laufwerke modifizieren. Das Programm fragt beim Aufruf alle angeschlossenen Laufwerke ab und zeigt Ihnen dann das entsprechende Menü an. Sollten alle drei Laufwerke angeschlossen sein, würde folgendes Bild erscheinen:

Programmaufruf

```
F1 - Partition Winchester drive 1
F2 - Partition Winchester drive 2
F3 - Partition Winchester drive 3

E - Exit to DOS

Enter selection <F1, F2, F3 or E>
```

Sie haben auch jetzt noch einmal die Möglichkeit, dieses Programm wieder zu verlassen. Möchten Sie fortfahren, wählen Sie das entsprechende Laufwerk aus, indem Sie die gewünschte *Funktionstaste* betätigen (nicht die einzelnen Tasten F und Ziffer). Es erscheint das Hauptmenü des Programmes, das je nach Zustand der Platte so aussieht wie im Bild 18.1 oder Bild 18.2 wiedergegeben:

```
   Partition        Start      End        Size in      Percentage
     Type          Cylinder   Cylinder   Kilobytes    of the Disk
   -----------     ---------  ---------  -----------  ------------
1. DOS (16)           0         615        21411         100.0%
2. Unallocated
3. Unallocated
4. Unallocated

Winchester unit = 0
Default boot partition = 1
Maximum cylinder Number = 615
Minimum DOS allocation = 32 Kilobytes

F1 - Select default boot partition
F2 - Allocate DOS partition
F3 - Delete DOS partition
E  - Exit

Enter selection <F1, F2, F3 or E >_
```

Bild 18.1
PART-Hauptmenü 1

	Partition Type	Start Cylinder	End Cylinder	Size in Kilobytes	Percentage of the Disk
	-----------	----------	----------	-------------	--------------
1.	non-DOS	0	615	21411	100.0%
2.	Unallocated				
3.	Unallocated				
4.	Unallocated				

Winchester unit = 0
Default boot partition = 1
Maximum cylinder Number = 615
Minimum DOS allocation = 32 Kilobytes

F1 - Select default boot partition
F2 - Allocate DOS partition
E - Exit

Enter selection <F1, F2 or E >_

Bild 18.2
PART-Hauptmenü 2

Das *Hauptmenü 1* erscheint, wenn Sie die Möglichkeit haben, wenigstens einen nichtbestimmten Teil als DOS-Teil zu bestimmen. Ebenso wenn wenigstens ein DOS-Teil (DOS oder DOS (16)) vorhanden ist, dessen Größe geändert werden kann **(F2)**. Mit der letzten Möglichkeit **(F3)** können Sie einen DOS-Teil wieder löschen.

Das *Menü 2* unterscheidet sich vom *Menü 1* dadurch, daß der erste Teil der Platte als Nicht-DOS-Teil bezeichnet ist. Diesen Teil können Sie mit dem Programm PART nicht bearbeiten.

Nicht-DOS-Teile können Sie mit der Taste **F3** nicht löschen.

Programmablauf

Das Hauptmenü ermöglicht Ihnen, zwischen drei Funktionen zu wählen:

- F1 – Bestimmung des Startlaufwerkes
- F2 – Aufteilung des Speicherplatzes in MS-DOS Bereiche
- F3 – Löschung eines MS-DOS zugeteilten Bereiches

Gleichzeitig wird angezeigt, wie die Platte momentan aufgeteilt ist. Nach jeder Eingabe wird das Menü entsprechend geändert. Das Hauptmenü zeigt folgende Informationen über die Aufteilung an:

- Art des zugeteilten Bereiches
- Größe des einzelnen Bereiches
- absoluten Speicherplatz auf der Platte für jeden Bereich
- Nummer des Laufwerkes
- Startlaufwerk

Die Art des zugeteilten Bereiches kann sein:

- *DOS*, dieser Teil ist für MS-DOS reserviert
- *DOS (16)*, dieser Teil ist für MS-DOS reserviert und benutzt im F.A.T. Einträge mit einer Größe von 16 Bit (s. a. Kapitel 11, FORMAT).
- *Non-DOS*, dieser Teil ist für ein anderes Betriebssystem reserviert und kann nicht bearbeitet werden
- *Unallocated*, dieser Teil ist noch nicht bestimmt und kann für MS-DOS reserviert werden

Programmablauf

Die folgenden Begriffe werden im Menü benutzt:

Start Cylinder	*Startzylinder* ist der erste Zylinder, der diesem Bereich zugewiesen ist.
End Cylinder	*Endzylinder* ist der letzte Zylinder, der diesem Bereich zugewiesen ist.
Size in Kilobytes	*Größe in Kilobytes*, gibt die Größe des Bereiches an.
Percentage of the disk	*Prozent der Platte* zeigt an, wieviel Prozent der Platte diesem Bereich zugeordnet sind.
Winchester unit	*Nummer des Laufwerkes* zeigt an, für welches Laufwerk Sie gerade die Änderungen vornehmen. Die Numerierung beginnt mit Null.
Default boot partition	*Startlaufwerk* zeigt an, welches logische Laufwerk Sie als Laufwerk für den Systemstart bestimmt haben.
Maximum cylinder Number	*Letzte Zylindernummer* zeigt die Gesamtzahl der verfügbaren Zylinder dieses Laufwerkes an.
Minimum DOS allocation	*Kleinster DOS-Teil* gibt die kleinste Größe an, die Sie für einen Teilbereich bestimmen dürfen.

PART

Programmablauf

Die letzten Zeilen des Menüs zeigen Ihnen Ihre Auswahlmöglichkeiten. Sie werden aufgerufen durch Betätigen der entsprechenden Funktionstaste oder des Buchstabens E. Folgende Möglichkeiten bestehen:

- F1 - Bestimmung des Startlaufwerkes
- F2 - Bestimmung eines DOS-Teils
- F3 - Löschung eines DOS-Teils
- E - Verlassen des Menüs und Rückkehr zur Laufwerkwahl

Bestimmung des Startlaufwerkes

Als Startlaufwerk sollten Sie das Laufwerk auswählen, mit dem Sie am meisten arbeiten. Dies kann jedes der vier Teile sein. Wird das System von der Winchester gebootet, ohne einen bestimmten Teil festzulegen, wird auf das Startlaufwerk zugegriffen. Dies wird meistens der Fall sein. Nähere Erklärungen dazu finden Sie im Kapitel 6.

Sie bestimmen das Startlaufwerk folgendermaßen:

- Rufen Sie das Programm PART auf.
- Wählen Sie ein Laufwerk aus

Es könnte folgendes Menü auf Ihrem Bildschirm erscheinen:

	Partition Type	Start Cylinder	End Cylinder	Size in Kilobytes	Percentage of the Disk
1.	DOS (16)	0	615	21411	100.0%
2.	Unallocated				
3.	Unallocated				
4.	Unallocated				

Winchester unit = 0
Default boot partition = 0
Maximum cylinder Number = 615
Minimum DOS allocation = 32 Kilobytes

F1 - Select default boot partition
F2 - Allocate DOS partition
F3 - Delete DOS partition
E - Exit

Enter selection <F1, F2, F3 or E >_

Programmablauf

Die Zeile Default boot partition zeigt eine 0. Dies soll jetzt geändert werden. Dafür betätigen Sie die Taste **F1**, und es erscheint in der letzten Zeile:

Select partition (1 - 4) or RETURN for none:_

Sie können jetzt mit der folgenden Eingabe das Startlaufwerk dieser Winchester bestimmen. Betätigen Sie nur die Taste **RETURN**, wird kein Startlaufwerk definiert. Nach der Eingabe wird der Bildschirm gelöscht, und es erscheint das Menü wieder. Dabei ist Ihre Eingabe berücksichtigt. Geben Sie eine **1** ein, erscheint folgendes Bild:

	Partition Type	Start Cylinder	End Cylinder	Size in Kilobytes	Percentage of the Disk
	-----------	----------	----------	-------------	--------------
1.	DOS (16)	0	615	21411	100.0%
2.	Unallocated				
3.	Unallocated				
4.	Unallocated				

Winchester unit = 0
Default boot partition = 1
Maximum cylinder Number = 615
Minimum DOS allocation = 32 Kilobytes

F1 - Select default boot partition
F2 - Allocate DOS partition
F3 - Delete DOS partition
E - Exit

Enter selection <F1, F2, F3 or E >_

Die Zeile Default boot partition wurde jetzt geändert. Dort steht eine 1.

Bestimmung eines Bereiches

Ein Bereich wird durch Angabe des *Start-* und *Endzylinders* bestimmt. Dabei wird vom Programm automatisch die *Größe in Kilobytes* errechnet, ebenso wie die entsprechende *Prozentzahl.* Sie können *nichtbestimmte Bereiche* (Unallocated) für DOS bestimmen und auch bereits bestimmte *DOS-Teile* (DOS oder DOS (16)) neu festlegen. Dabei kann ein Zylinder nicht für mehrere Teilbereiche bestimmt werden. Die maximale Größe eines Teilbereiches darf 32 Mb betragen.

Sie bestimmen einen DOS-Teil folgendermaßen:

- Rufen Sie das Programm PART auf.

- Wählen Sie ein Laufwerk aus.

Es könnte folgendes Menü auf Ihrem Bildschirm erscheinen:

```
    Partition        Start          End          Size in       Percentage
      Type          Cylinder      Cylinder      Kilobytes      of the Disk
    ----------     ----------    ----------   -------------  -------------
1.  DOS (16)           0            615          21411          100.0%
2.  Unallocated
3.  Unallocated
4.  Unallocated

Winchester unit = 0
Default boot partition = 1
Maximum cylinder Number = 615
Minimum DOS allocation = 32 Kilobytes

F1 - Select default boot partition
F2 - Allocate DOS partition
F3 - Delete DOS partition
E  - Exit

Enter selection <F1, F2, F3 or E >_
```

Programmablauf

Betätigen Sie jetzt die Taste **F2**, und es erscheint in der letzten Zeile

Select Partition (1 - 4):_

In diesem Beispiel soll das Winchesterlaufwerk in mehrere einzelne Laufwerke aufgeteilt werden. Auf die Frage nach dem zu *bearbeitenden Teil* (Select partition) geben Sie jetzt eine **1** ein.

Es erscheint folgende neue Zeile:

Start Cylinder:_

Das erste Laufwerk soll 200 Zylinder groß werden. (Es ist auch möglich, einen anderen Teil hier auszuwählen oder eine andere Größe festzulegen.) Bestimmen Sie hier den ersten Zylinder des Bereiches mit Null. Geben Sie **0** ein. Bei Eingabe einer nicht erlaubten Zahl ertönt der Summer und die Eingabe wird nicht angenommen. Haben Sie die Eingabe mit **RETURN** abgeschlossen, erscheint in der folgenden Zeile:

End Cylinder:_

Bestimmen Sie jetzt den letzten Zylinder des Bereiches mit **199**. In diesem Beispiel sollte der erste Bereich 200 Zylinder groß werden. Bevor Sie jetzt die Eingabe abschließen, sehen Sie sich den Bildschirm an. Er sollte so aussehen:

```
     Partition      Start       End        Size in     Percentage
       Type        Cylinder   Cylinder    Kilobytes   of the Disk
    -----------   ----------  ----------  ----------  ------------
 1. DOS (16)          0          615        21411        100.0%
 2. Unallocated
 3. Unallocated
 4. Unallocated

Winchester unit = 0
Default boot partition = 1
Maximum cylinder Number = 615
Minimum DOS allocation = 32 Kilobytes

Enter selection <F1, F2, F3 or E >
Select Partition (1 - 4):1
Start Cylinder: 0
End Cylinder: 199_
```

Nach Betätigen der Taste **RETURN** wird der Schirm gelöscht, und es erscheint folgendes Bild:

```
     Partition      Start       End        Size in     Percentage
       Type        Cylinder   Cylinder    Kilobytes   of the Disk
    -----------   ----------  ----------  ----------  ------------
 1. DOS               0          199         6963         32.5%
 2. Unallocated
 3. Unallocated
 4. Unallocated

Winchester unit = 0
Default boot partition = 1
Maximum cylinder Number = 615
Minimum DOS allocation = 32 Kilobytes

F1 - Select default boot partition
F2 - Allocate DOS partition
F3 - Delete DOS partition
E  - Exit

Enter selection <F1, F2, F3 or E>
```

Programmablauf

Wie Sie sehen, ist Ihre Eingabe berücksichtigt. Ebenso ist automatisch die Größe des gewählten Bereiches errechnet sowie der entsprechende Prozentsatz. Bestimmen Sie einen Bereich, der größer als 15 Mb oder 451 Zylinder ist, bezeichnet PART diesen Teil als DOS (16). Damit wird das Betriebssystem angewiesen, im F.A.T. dieses Teiles alle Einträge mit 16 Bit vorzunehmen. Mit den bisherigen 12-Bit-Einträgen konnten nur Teilbereiche von maximal 15 Mb verwaltet werden.

Fehlermeldungen

Bestimmen Sie einen Teilbereich und geben richtige Werte ein, erscheint nach der letzten Eingabe ein neues Menü. Ihre Eingaben sind bereits berücksichtigt worden. Geben Sie falsche Werte ein, werden diese entweder nicht angenommen (es ertönt der Summer), oder es erscheint eine der folgenden Fehlermeldungen:

Error - Requested allocation overlaps DOS partition	Fehler - eingegebener Bereich überschneidet DOS-Teil

ERKLÄRUNG:
Sie haben für den gerade bestimmten Teil Zylindernummern eingegeben, die schon für einen anderen Teil vergeben sind. Bestimmen Sie den Bereich neu.

Error - Requested allocation overlaps non-DOS partition	Fehler - eingegebener Bereich überschneidet Nicht-DOS-Teil

ERKLÄRUNG:
Sie haben für den gerade bestimmten Teil Zylindernummern eingegeben, die schon für den Nicht-DOS-Teil vergeben sind. Dieser Teil kann von PART nicht bearbeitet werden. Bestimmen Sie den Bereich neu.

Error - Requested allocation is less than required DOS minimum.	Fehler - eingegebener Bereich ist kleiner als das DOS-Minimum

ERKLÄRUNG:
Durch die eingegebenen Zylindernummern wird ein Bereich definiert, der kleiner ist als 32Kb. Dies ist das absolute Minimum für einen DOS-Teil.

Error - Requested allocation exceeds DOS maximum of 32 megabytes	Fehler - eingegebener Bereich ist größer als das DOS-Maximum

ERKLÄRUNG:
Durch die eingegebenen Zylindernummern wird ein Bereich definiert, der größer ist als 32Mb. Dies ist das absolute Maximum für einen DOS-Teil. Durch den Winchesterkontroller kann nicht mehr verwaltet werden.

Programmablauf

Löschen eines Bereiches

Genauso wie Sie einen Bereich der Platte für DOS bestimmen, können Sie einen für DOS bestimmten Teil wieder löschen. Mit diesem Teil des Programmes können Sie keinen Nicht-DOS-Teil löschen. Auch ist es nicht möglich, einen nicht bestimmten Teil zu löschen.

Sie löschen einen DOS-Teil folgendermaßen:

- Rufen Sie das Programm PART auf.
- Wählen Sie ein Laufwerk aus.

Es könnte folgendes Menü auf Ihrem Bildschirm erscheinen:

	Partition Type	Start Cylinder	End Cylinder	Size in Kilobytes	Percentage of the Disk
1.	DOS	0	100	3433	16.0%
2.	DOS	101	250	5222	24.3%
3.	DOS	251	303	1810	8.4%
4.	DOS	304	615	10828	50.4%

```
Winchester unit = 0
Default boot partition = 1
Maximum cylinder Number = 615
Minimum DOS allocation = 32 Kilobytes

F1 - Select default boot partition
F2 - Allocate DOS partition
F3 - Delete DOS partition
E  - Exit

Enter selection <F1, F2, F3 or E >_
```

Betätigen Sie jetzt die Taste F3 und es erscheint

Select partition (1 - 4):_

Geben Sie jetzt die Zahl des Bereiches ein, den Sie löschen wollen, z. B. eine 3. Dann erscheint folgendes Bild:

	Partition Type	Start Cylinder	End Cylinder	Size in Kilobytes	Percentage of the Disk
	----------	----------	----------	----------	----------
1.	DOS	0	100	3433	16.0%
2.	DOS	101	250	5222	24.3%
3.	DOS	251	303	1810	8.4%
4.	Unallocated				

Winchester unit = 0
Default boot partition = 1
Maximum cylinder Number = 615
Minimum DOS allocation = 32 Kilobytes

F1 - Select default boot partition
F2 - Allocate DOS partition
F3 - Delete DOS partition
E - Exit

Enter selection <F1, F2, F3 or E >_

Der gelöschte Teil ist jetzt komplett verschwunden. Die anderen Teile sind dadurch nicht automatisch größer geworden. Wollen Sie die freie Kapazität nutzen, können Sie jetzt die beiden restlichen Teile vergrößern. Haben Sie den letzten DOS-Teil gelöscht, erscheint in der Auswahl die Zeile F3 nicht mehr.

Programmablauf

Haben Sie alle gewünschten Änderungen vorgenommen, geben Sie im Hauptmenü ein **E** ein. Sie verlassen damit das Programm und es erscheint folgendes Bild:

```
F1 - Partition Winchester drive 1

M - Make changes, exit to MFM-200
A - Abort all changes, exit to DOS

Enter selection <F1, M or A>_
```

Es ist möglich, an Ihren Z-200 PC bis zu drei Winchesterlaufwerke anzuschließen. Obwohl in den meisten Fällen dies nicht voll ausgenutzt wird, können Sie mit diesem Programm alle drei Laufwerke modifizieren. Das Programm fragt beim Aufruf alle angeschlossenen Laufwerke ab und zeigt Ihnen dann das entsprechende Menü an. Sollten alle drei Laufwerke angeschlossen sein, würde folgende Auswahl erscheinen:

```
F1 - Partition Winchester drive 1
F2 - Partition Winchester drive 2
F3 - Partition Winchester drive 3


M - Make changes, exit to MFM-200
A - Abort all changes, exit to DOS

Enter selection <F1, F2, F3 M or A>_
```

Sie haben jetzt die Möglichkeit, wieder ein Winchesterlaufwerk aufzurufen und weitere Änderungen vorzunehmen (**F1 - F3**). Sie können jetzt auch das Programm verlassen, ohne daß die Änderungen gespeichert werden (**A**). Sollen die eingegebenen Änderungen gespeichert werden, geben Sie ein **M**(ake changes) ein. Nach einem kurzen Moment erscheint das Prompt des Betriebssystemes, und Sie haben das Programm endgültig verlassen. Damit die Änderungen vom System auch erkannt werden, ist es jetzt erforderlich, das System neu zu starten.

Tabelle der logischen Laufwerke

Diese Tabelle enthält alle *Informationen* über die vier bezeichneten Laufwerke, in die die Platte aufgeteilt werden kann. Sie enthält die folgenden Informationen:

- Boot-Byte
- Startzylinder, Kopf & Sektor
- Endzylinder, Kopf & Sektor
- Betriebssystem
- Anzahl der dem Teil vorausgehenden Sektoren
- Anzahl der vom Teil belegten Sektoren

Die Tabelle 18.1 zeigt eine Übersicht.

Die Tabelle wird am Ende des *Boot-Satzes* im Sektor 1, Zylinder 0, Kopf 0 auf der Winchester gespeichert. Der *Boot-Satz* ist ein spezielles Programm für Winchesterlaufwerke. Es enthält Informationen über das Startlaufwerk der Winchester. Soll von der Winchester gebootet werden, wird der Boot-Satz in den Arbeitsspeicher geladen.

In der Tabelle 18.1 werden folgende Begriffe benutzt:

Boot Durch das *Boot-Byte* wird das Startlaufwerk definiert und der Boot-Satz greift auf diesen Teil der Platte zu. Wurde mit dem Programm PART ein Startlaufwerk bestimmt, enthält das Boot-Byte die Nummer des logischen Laufwerkes.

Sys Das *System-Byte* definiert die Art des Betriebssystemes auf diesem Teil. Das Programm PART kann nur Teile der Platte bearbeiten, deren System-Byte 0 oder 1 ist. Nicht definierte Teile werden mit 0 gekennzeichnet, DOS- oder DOS-(16)-Teile werden mit 1 gekennzeichnet. Alle anderen Werte definieren Nicht-DOS-Teile.

Tabelle der logischen Laufwerke

Start — Das Feld des *Startwertes* definiert den Kopf (H), Sektor (S) und Zylinder (Cyl), an dem dieser Teil auf der Platte beginnt.

Ende — Das Feld des *Endwertes* definiert den Kopf (H), Sektor (S) und Zylinder (Cyl), an dem dieser Teil auf der Platte endet.

rel. sec. — Das Feld des *Relativwertes* enthält die Anzahl der Sektoren, die diesem Teil auf der Platte vorangehen. Das Feld besteht aus 4 Bytes (32 Bits). Es werden die beiden niedrigwertigen Bytes jeweils zuerst gespeichert.

num. secs. — Das Feld des *Nummernwertes* enthält die Anzahl der Sektoren, die zu diesem Teil gehören. Das Feld besteht ebenfalls aus 4 Bytes.

Die relativen Speicherplätze 1FE und 1FF des ersten Sektors zeigen an, ob eine Tabelle vorhanden ist. Dies ist der Fall bei den Werten 55h für 1FE und AAh für 1FF.

Das Programm Part prüft ebenfalls das letzte Byte des Boot-Satzes auf seine Richtigkeit. Ist der Wert korrekt, wird die vorhandene Tabelle von PART benutzt. Ist der Wert nicht korrekt, wird vom Programm automatisch eine neue Tabelle erstellt.

Struktur der Tabelle

Die Tabelle kann Informationen für maximal vier logische Laufwerke speichern. Die Tabelle 18.1 zeigt die Struktur des 16-Byte-Feldes. Die Startadresse (innerhalb des Boot-Satzes) für jedes Laufwerk ist:

- 01BE für Teilbereich #1
- 01CE für Teilbereich #2
- 01DE für Teilbereich #3
- 01EE für Teilbereich #4

Tabelle 18.1 Struktur des Eintrages

OFFSET	BYTES	NAME	ERKLÄRUNG
00	1	Boot	Boot-Byte
01	1	Start	erster Kopf des Bereiches
02	1		erster Sektor des Bereiches
03	1		erster Zylinder des Bereiches
04	1	Sys	System-Byte
06	1	Ende	letzter Kopf des Bereiches
06	1		letzter Sektor des Bereiches
07	1		letzter Zylinder des Bereiches
08	4	rel.sec.	Anzahl der dem Teilbereich vorausgehenden Sektoren
0C	4	num.sec.	Größe des Teilbereichs in Sektoren

Fehlermeldungen

Cannot read the boot sector	Der Boot-Sektor kann nicht gelesen werden

ERKLÄRUNG:
Der erste Sektor der Platte ist defekt. Initialisieren Sie die Platte neu. Tritt der Fehler wieder auf, wenden Sie sich an Ihren Händler.

Cannot write the boot sector	Der Boot-Sektor kann nicht beschrieben werden

ERKLÄRUNG:
Es wird versucht, Änderungen auf die Platte zu schreiben. Der erste Sektor ist fehlerhaft. Initialisieren Sie die Platte neu. Tritt der Fehler wieder auf, wenden Sie sich an Ihren Händler.

Error - Disk is fully allocated to non-DOS partitions, no partitioning is possible	Die Platte ist komplett für Nicht-DOS-Teile eingerichtet. Eine weitere Aufteilung ist nicht möglich

ERKLÄRUNG:
Nachdem eine Winchester ausgewählt wurde, stellt das Programm fest, daß die gesamte Platte mit Nicht-DOS-Teilen belegt ist. Das Programm kann diese Bereiche nicht ändern. Wählen Sie eine andere Platte aus.

Error - Can not communicate with the Winchester controller	Winchesterkontroller nicht vorhanden

ERKLÄRUNG:
Das Programm findet den Winchesterkontroller nicht. Es ist möglich, daß die Karte nicht richtig installiert ist. Überprüfen Sie alle Anschlüsse und starten Sie das Programm neu. Erscheint der Fehler wieder, ist die Karte wahrscheinlich defekt.

Fehlermeldungen

Error - No winchester drives attached	Es sind keine Winchester-laufwerke angeschlossen

ERKLÄRUNG:
Das Programm PART fragt bei seinem Aufruf alle vorhandenen Laufwerke ab. Haben Sie keine Winchesterlaufwerke angeschlossen, kann das Programm nicht weiter ablaufen und wird sofort mit dieser Fehlermeldung beendet.

ASGNPART

Übersicht

Das Programm ASGNPART verbindet den physikalischen Teil des Winchesterlaufwerkes mit einem logischen Laufwerknamen. Das Programm befindet sich auf der ersten Systemdiskette.

Während des Systemstartes gibt MS-DOS automatisch dem ersten Teil der Winchester einen Laufwerknamen. Dies ist in Abhängigkeit von der Anzahl der Diskettenlaufwerke der Buchstabe C, D oder E. Bezeichnet wird entweder der erste Bereich der Winchester oder der Teil, von dem aus gestartet wurde.

Haben Sie mit Hilfe des Programmes CONFIGUR das System auf automatische Bezeichnung gestellt (s. a. Kapitel 11), können Sie gleichzeitig immer nur auf einen Teilbereich zugreifen. Dies ist für die Programme TOPVIEW und MS-NET erforderlich.

Haben Sie das System auf manuelle Bezeichnung eingestellt, können Sie für alle definierten Teilbereiche Laufwerkbezeichnungen vergeben. Erlaubte Eingaben zeigt der Hilfsbildschirm.

Programmaufruf

Syntax

```
[d:Weg] ASGNPART [?]
[d:Weg] ASGNPART u:
[d:Weg] ASGNPART u:p D:
```

dabei bedeutet:

?	zeigt das Menü des Programmes
d:	vor dem Programmnamen bestimmt das aufzurufende Laufwerk
Weg	vor dem Programmnamen bestimmt das aufzurufende Verzeichnis
u:	Nummer des Winchesterlaufwerkes (0–3)
p:	Nummer des Teilbereiches der Winchester, das bezeichnet werden soll (1 - 4)
D:	Name des Laufwerkes, mit dem der Teilbereich bezeichnet werden soll (C - P)

Bemerkungen

Sie können Ihr *Winchesterlaufwerk* in maximal vier Bereiche aufteilen. Die Teile werden mit den Ziffern 1 bis 4 bezeichnet. Jeder Teil kann ein anderes Betriebssystem enthalten. Damit Sie auf die einzelnen Bereiche unter MS-DOS zugreifen können, müssen die Bereiche mit Laufwerknamen bezeichnet werden. Diese *Verbindung* zwischen Winchesterteil und Betriebssystem wird durch das Dienstprogramm ASGNPART hergestellt.

Haben Sie Ihren Z-200 PC auf automatischen Systemstart vom Winchesterlaufwerk aus eingestellt, wird der Bootteil der Winchester vom Betriebssystem mit C: bezeichnet. Alle anderen Teile müssen von Ihnen mit dem Programm ASGNPART benannt werden.

ACHTUNG:
Benutzen Sie dieses Programm nicht für Laufwerke, denen Sie mit dem Befehl ASSIGN vorübergehend einen anderen Namen zugewiesen haben. Durch die Umbenennung wird nicht auf das ursprüngliche Laufwerk zurückgegriffen, sondern auf das umgeleitete. Dadurch können die wahren Eigenschaften des Laufwerkes verdeckt werden und das Programm arbeit nicht fehlerfrei. Ebenso ist es nicht erlaubt, für das Netzwerk bestimmte Laufwerke zu bezeichnen.

Da sich das *Winchesterlaufwerk* in den meisten Fällen wie ein Diskettenlaufwerk verhält, kann man die Aufgabe des Programmes ASGNPART mit dem Einschieben einer Diskette in das Laufwerk vergleichen. So wie die Diskette erst in das Laufwerk gelegt werden muß, bevor die Daten gelesen werden können, muß der *Winchesterteil* erst bezeichnet werden, bevor die Daten gelesen werden können.

Programmablauf

Sie können das Programm ASGNPART mit verschiedenen Befehlen aufrufen. Mit Eingabe des Fragezeichens rufen Sie das Hauptmenü des Programmes auf. Geben Sie ein

```
ASGNPART ?
```

und es erscheint folgendes Bild:

ASGNPART

Programmablauf

```
ASGNPART Version 3.03
Copyright 1985, Zenith Data Systems Corporation

The ASGNPART utility is used to make a logical connection
between the partitions of a winchester disk drive and the
drive names C:, D:, E:, and F:

Usage: ASGNPART[?][u:][u:p d:]

Where:

  ?    - Print this list
  u:   - Unit number of winchester (0 - 7)
  p    - Partition number to ASGNPART (1 - 4)
  d:   - Drive letter (C: - F:)

If only [u:] is specified, a table of valid partitions
for drive [u:] is printed.
```

Auf Ihrem Bildschirm kann eine leicht veränderte Meldung erscheinen. Haben Sie ein weiteres Diskettenlaufwerk oder ein zusätzliches Winchesterlaufwerk angeschlossen, wird dieses vom Programm vorher abgefragt und bei der Bildschirmmeldung entsprechend berücksichtigt.

Den momentanen Status des Winchesterlaufwerkes bekommen Sie angezeigt, wenn Sie den Befehl mit der *Laufwerksnummer* eingeben. Es erscheint dann eine Tabelle, die die Größe der einzelnen Teilbereiche anzeigt. Geben Sie für das erste angeschlossene Laufwerk ein:

```
ASGNPART 0:
```

Es könnte z.B. folgende Tabelle erscheinen:

```
                 ASGNPART Version 3.01
Copyright 1985, Zenith Data Systems Corporation

     Partition        Start        End         Size in
       Type          Cylinder     Cylinder     Kilobytes
    ----------      ----------   ----------   ----------
1. DOS                  0           230          7853
2. DOS                231           303          2482
3. Unallocated
4. Unallocated

  Drive C: = 0:1
  Drive D: = Unassigned
  Drive E: = Unassigned
  Drive F: = Unassigned
```

Diese Tabelle gleicht der vom Programm PART erstellten Tabelle.

Der zweite Teilbereich kann jetzt mit dem Namen **D:** bezeichnet werden. Danach erst ist es möglich, auf die dort gespeicherten Daten zuzugreifen. Geben Sie ein:

```
ASGNPART 0:2 D:
```

Das Programm bestätigt mit der Meldung:

DOS partition 2 assigned to drive d	DOS-Teil 2 bezeichnet als Laufwerk D

Bei manueller Bezeichnung kann der Teilbereich anschließend nicht neu benannt werden. Bei automatischer Bezeichnung kann die gerade vergebene Laufwerkbezeichnung durch erneuten Programmaufruf für einen anderen Teilbereich vergeben werden.

Programmablauf

Nachdem der Teil des Winchesterlaufwerkes mit einem Namen bezeichnet wurde, kann dieser Teil von Ihnen wie ein normales Diskettenlaufwerk behandelt werden. Es sind jedoch die folgenden Regeln zu beachten:

- Der Teilbereich darf nicht kleiner als 32Kb oder größer als 32Mb sein.
- Ein Teilbereich kann nur mit einem Namen bezeichnet werden.
- Die Nummer des Winchesterlaufwerkes (u:) muß zwischen 0 und 7 betragen.
- Die Nummer des Teilbereiches (p) muß zwischen 1 und 4 betragen.
- Die Laufwerkbezeichnunng darf nur aus den angezeigten Buchstaben ausgewählt werden. Das Hauptmenü zeigt die gültigen Buchstaben an.

Fehlermeldungen

Cannot communicate with Winchester controller	Winchesterkontroller nicht vorhanden

ERKLÄRUNG:
Es ist kein Winchesterlaufwerk an Ihrem Zenith-PC angeschlossen. Das Programm kann nicht ablaufen.

Cannot read partition table	Die Partition-Tabelle kann nicht gelesen werden

ERKLÄRUNG:
Sie haben versucht, einen nicht vorhandenen Teil des Winchesterlaufwerkes zu bezeichnen. Überprüfen Sie Ihre Eingabe.

Inalid drive/ drive not available	Falsche Laufwerkbezeichnung; Bezeichnung nicht vorhanden

ERKLÄRUNG:
Sie haben versucht, einen Laufwerknamen einzugeben, der nicht in der Liste aufgeführt war. Überprüfen Sie Ihre Eingabe.

Invalid partition selection	Falsche Bezeichnung des Bereiches

ERKLÄRUNG:
Es sind maximal 4 Bereiche vorhanden, die mit den Ziffern 1 bis 4 bezeichnet werden. Sie haben eine falsche Ziffer eingegeben. Überprüfen Sie Ihre Eingabe.

ASGNPART

Fehlermeldungen

Invalid version of IO.SYS	Falsche Version der Datei IO.SYS

ERKLÄRUNG:
Sie benutzen das Programm ASGNPART unter einer falschen Version des Betriebssystems. Das Programm ASGNPART mit den Versionen 3.XX arbeitet nur unter den DOS-Versionen 3.XX.

Invalid Winchester drive number given	Falsche Nummer des Winchesterlaufwerkes eingegeben

ERKLÄRUNG:
Sie haben für das Winchesterlaufwerk eine falsche Ziffer eingegeben. Erlaubt ist eine Ziffer von 0 bis 7, die aber nicht größer sein darf, als die Anzahl der angeschlossenen Laufwerke. Überprüfen Sie Ihre Eingabe.

Not a DOS partition	Kein DOS-Teil

ERKLÄRUNG:
Unter MS-DOS können nur Teilbereiche bezeichnet werden, die von MS-DOS vorher festgelegt wurden (s. a. Kapitel 18). Diese Teile werden in der Übersicht mit DOS oder DOS 16 benannt.

Partition is already in use	Der Bereich ist bereits benannt

ERKLÄRUNG:
Sie haben versucht, einen Teilbereich erneut zu benennen. Das ist nur möglich, wenn vorher das System neu gestartet wird oder die Zuteilung der Laufwerke automatisch vorgenommen wird (s. a. CONFIGUR).

Fehlermeldungen

Partition is too large — Der Bereich ist zu groß

ERKLÄRUNG:
Sie haben versucht, einen Teilbereich zu benennen, der größer ist als 32Mb. Dies ist nicht möglich, da der Winchesterkontroller keine Bereiche in dieser Größe verwalten kann. Teilen Sie den Bereich mit dem Programm PART.

Partition is too small — Der Bereich ist zu klein

ERKLÄRUNG:
Sie haben versucht, einen Teilbereich zu benennen, der kleiner ist als 32Kb. Dies ist nicht erlaubt (s. a. Kapitel 18).

Kapitel 20

SHIP

Übersicht

Dieses Programm bewegt die Schreib-Lese-Köpfe Ihres Winchesterlaufwerkes in eine Position, wo sie während des Transportes keine Daten zerstören können. Dieses Programm ist auf der zweiten Systemdiskette enthalten. Rufen Sie es immer auf, bevor Sie Ihren Zenith PC transportieren.

Programmaufruf

Syntax

[d:] [Weg] SHIP

dabei bedeutet:

d: vor dem Programmnamen bestimmt das aufzurufende Laufwerk

Weg vor dem Programmnamen bestimmt das aufzurufende Verzeichnis

Bemerkungen

Winchesterlaufwerke sind hochempfindliche Geräte, die durch leichte Vibrationen bereits Schaden nehmen können. Die Daten auf Ihrer Winchesterplatte sind ebenfalls stark gefährdet. Sie können leicht zerstört werden. Daher sollten Sie besondere Vorsichtsmaßnahmen treffen, um die Daten zu schützen. Sie sollten bereits daran denken, wenn Sie Ihr Gerät im Raum transportieren.

Durch das Programm SHIP werden Laufwerk und Daten geschützt. Das Programm fährt die Schreib-Lese-Köpfe in die Mitte der Platte. Hier sind die Köpfe nicht so gefährdet. Sollten durch den Transport die Platten in Bewegung geraten, schwingen sie in der Mitte nicht so sehr wie am Rand. Daher werden die Platten nicht so schnell mit den Köpfen in Berührung kommen. Die Gefahr eines Head-Crash ist damit nicht mehr so groß. Die inneren Zylinder der Platte werden deshalb auch *Transportzylinder* (Shipping Cylinder) genannt.

Der Winchesterkontroller fährt die Schreib-Lese-Köpfe nach dem Einschalten des Gerätes immer auf den Zylinder 0. Daher bleiben die Köpfe nur solange über dem Transportzylinder, bis das Gerät wieder eingeschaltet wird.

Programmablauf

Rufen Sie das Programm mit seinem Namen auf. Geben Sie ein

SHIP

und es erscheint folgende Meldung auf dem Bildschirm:

SHIP Version 3.01
Copyright 1985 (C), Zenith Data Systems Corporation.

The SHIP utility helps you to:

* Position the read/write heads of the Winchester disk at a safe location for subsequent transportation of the disk unit.

After running SHIP, no further operation on your computer will be possible. At that time, turn off the power to the computer and prepare it for shipping.

Do you wish to continue (Y/N):

Der Text erklärt noch einmal die Arbeitsweise des Programmes und weist darauf hin, daß nach Programmende kein weiterer Zugriff auf die Platte mehr möglich ist. Geben Sie auf die Frage ein **N**(ein) ein, verlassen Sie das Programm und sind wieder im Betriebssystem. Geben Sie ein **Y** für Ja ein, werden die Köpfe zum Transportzylinder gefahren und es erscheint folgender Text:

Heads moved to shipping cylinder. Turn off your microcomputer and prepare it for shipping.

Die Köpfe sind über dem Transportzylinder. Schalten Sie das Gerät jetzt aus.

Damit ist das Programm beendet. Es kann keine Funktion mehr aufgerufen werden.

Fehlermeldung

Unable to communicate with the Winchester controller	Winchesterkontroller nicht vorhanden

ERKLÄRUNG:
Das Programm findet den Winchesterkontroller nicht. Es ist möglich, daß die Karte nicht richtig installiert ist. Überprüfen Sie alle Anschlüsse und starten Sie das Programm neu. Erscheint der Fehler wieder, ist die Karte wahrscheinlich defekt.

Kapitel 21

DETECT

Übersicht

Dieses Dienstprogramm untersucht die Winchesterlaufwerke Ihres Zenith PC auf fehlerhafte Sektoren und markiert sie. Damit können die Sektoren von MS-DOS nicht mehr benutzt werden. Das Programm ist auf der zweiten Systemdiskette enthalten.

Mit dem Programm PREP wird beim Initialisieren der Platte bereits überprüft, welche Sektoren auf Grund eines Fehlers nicht benutzt werden können. Das Programm DETECT ergänzt diese Arbeit. Sollten in der Zwischenzeit weitere Sektoren beschädigt worden sein, werden sie jetzt festgestellt und für den weiteren Gebrauch gesperrt. Sämtliche fehlerhaften Sektoren werden in eine Tabelle eingetragen, die von PREP angelegt wurde. In die Tabelle können maximal 168 Adressen von Sektoren eingetragen werden.

Werden mit dem Programm DETECT fehlerhafte Sektoren gefunden, ist es notwendig, die Platte neu zu formatieren.

Programmaufruf

Syntax

[d:] [Weg] DETECT

d: vor dem Programmnamen bestimmt das aufzurufende Laufwerk

Weg vor dem Programmnamen bestimmt das aufzurufende Verzeichnis

Bemerkungen

Fehlerhafte Sektoren sind Stellen auf der Plattenoberfläche, die Daten nicht richtig speichern und dadurch eine Fehlermeldung hervorrufen. Diese Fehlermeldung wird ausgegeben, nachdem mehrere Male versucht wurde, auf die Platte zuzugreifen. Nach dieser Fehlermeldung müssen Sie den Vorgang abbrechen. Die Daten, die für diesen Sektor bestimmt waren, sind dann höchstwahrscheinlich verloren.

Das Dienstprogramm DETECT findet diese fehlerhaften Sektoren und trägt sie in eine Tabelle ein. Damit können diese Sektoren für die spätere Benutzung gesperrt werden und verursachen keine weiteren Datenverluste mehr. Die Tabelle der fehlerhaften Sektoren wurde bereits vom Programm PREP angelegt, als die Platte initialisiert wurde. In diese Tabelle können maximal 168 Sektoren eingetragen werden. Danach erscheint eine entsprechende Fehlermeldung.

Wird die gesamte Platte oder Teile davon neu formatiert, werden vom Programm FORMAT die fehlerhaften Sektoren entsprechend markiert. Auf sie kann dann in Zukunft nicht mehr zugegriffen werden.

Programmaufruf

Fehlerhafte Sektoren können durch folgende Ursachen hervorgerufen werden:

- starker Stoß der Platte (mehr als 5 G, für mehr als 11 Millisekunden)
- Fremdkörper im Laufwerkgehäuse (z. B. Staub)
- Fehlsteuerung durch die Kontrollerkarte
- Stromausfall des Laufwerkes

In diesen Fällen findet das Programm DETECT nicht immer den beschädigten Sektor. Sichern Sie dann alle Dateien der Platte und initialisieren Sie die Platte anschließend neu. Sollten Sie auch damit die beschädigten Sektoren nicht finden, wenden Sie sich an Ihren Händler.

Das Programm DETECT zerstört *keine* Daten, während es die Platte untersucht. Damit die gefundenen Sektoren gesperrt werden, muß die Platte anschließend neu formatiert werden. Dafür ist es erforderlich, sämtliche Dateien vorher auf Diskette zu kopieren. Anschließend können sie dann auf die neu formatierte Platte wieder zurückübertragen werden.

Programmablauf

Rufen Sie das Programm auf durch Eingabe von:

DETECT

Auf dem Bildschirm erscheint folgender Text:

```
                DETECT Version 3.01
Copyright (C) 1985, Zenith Data Systems Corporation

The DETECT utility helps you to:

   * Locate sectors that have failed since you last ran PREP

Do you wish to proceed with DETECT (Y/N)?
```

Der Text erklärt noch einmal die Arbeitsweise des Programmes. Geben Sie auf die Frage ein **N**(ein) ein, verlassen Sie das Programm und sind wieder im Betriebssystem. Geben Sie ein **Y** für Ja ein, erscheint die Frage nach der Laufwerknummer:

Winchester drive unit number (0 -3):_	Nummer des Winchester- laufwerkes (0 -3):_

Geben Sie hier die Nummer des zu untersuchenden Winchesterlaufwerkes ein (das erste Laufwerk wird mit Null bezeichnet). Danach erscheint die Frage nach bereits bekannten fehlerhaften Sektoren:

Enter bad sector adress, or zero to end:_	Geben Sie die Adresse des fehler- haften Sektors ein, oder Null zum Programmstart:

Wird während eines Zugriffs auf die Platte ein fehlerhafter Sektor festgestellt, gibt das Betriebssystem eine Fehlermeldung aus, die sich von der Meldung für Disketten leicht unterscheidet. Für Winchesterlaufwerke erscheint folgender Text:

Art error i/o-action drive X Sector adress of error is *nnnn* Abort, Retry, Ignore:	Fehler während Ein- / Ausgabe auf Laufwerk X, Adresse des fehler- haften Sektors ist nnnn Abbrechen, Wiederholen, Ignorieren:

Programmablauf

Dabei kann die *Art* des Fehlers folgende sein:

WRITE PROTECT	Diskette schreibgeschützt
BAD UNIT	Falsche Einheit
NOT READY	Laufwerk nicht bereit
BAD COMMAND	Falscher Befehl
DATA	Datenfehler
BAD CALL FORMAT	Falscher Aufruf
SEEK	Suchfehler
NON DOS-DISK	Keine DOS-Diskette
SECTOR NOT FOUND	Sektor nicht gefunden
WRITE FAULT	Schreibfehler
READ FAULT	Lesefehler
DISK	Plattenfehler

Die Ein- / Ausgabe bezieht sich auf das Lesen (READING) oder Schreiben (WRITING). Mit *X* wird das Laufwerk bezeichnet, daß den Fehler verursacht hat.

Die *logische Adresse* des fehlerhaften Sektors wird in hexadezimal (*nnnn*) angegeben. Die logische Adresse des gesamten Winchesterlaufwerkes beginnt mit dem ersten absoluten Sektor bei 0000h.

Erscheint diese Fehlermeldung während eines Programmes, sollten Sie die angezeigte Adresse notieren. Sie können versuchen, den Fehler mit der Eingabe von **R** zu beheben (gelingt dies nicht, muß der gesamte Vorgang wiederholt werden). Notieren Sie sich ebenfalls das Laufwerk und den Teil des Laufwerkes, wo der Fehler auftritt. Dieser Wert wird eventuell benötigt für die Fehlermeldung:

Invalid HEX value, Try again:	Falscher Hexadezimalwert, geben Sie neu ein:

Auf die Frage nach bekannten fehlerhaften Sektoren geben Sie jetzt die hier notierte Adresse(n) ein. Es erscheint dann die Frage erneut für die nächste Adresse. Haben Sie alle Adressen eingegeben, geben Sie eine **0** ein. Damit wird das Programm veranlaßt, die gesamte Platte nach weiteren Sektoren abzusuchen.

Programmablauf

Das Programm gibt jetzt die folgende Meldung aus:

Beginning detection...	Überprüfung...

und schreibt nach Ablauf die Zeile zu Ende:

Beginning detection... Completed	Überprüfung... beendet

Wurden keine weiteren Sektoren mehr gefunden, erscheint die Meldung:

No bad sectors detected.	Keine fehlerhaften Sektoren gefunden.

Wurden Sektoren gefunden, wird dies ebenfalls mitgeteilt:

Bad sectors located.	Fehlerhafte Sektoren gefunden.

Ebenso wird angezeigt, wenn die Tabelle korrigiert worden ist:

Tables modified.	Tabelle berichtigt.

Werden mehr als insgesamt 168 fehlerhafte Sektoren gefunden, kann das Programm nicht mehr ordnungsgemäß beendet werden, und es erscheint die Fehlermeldung:

Bad sector count exceeded for this drive.	Anzahl der fehlerhaften Sektoren ist zu groß für diese Platte.

Sie sollten dann die Platte neu initialisieren.

Wurde das Programm ordnungsgemäß beendet, ist es jetzt erforderlich, von der Diskette neu zu booten und die Winchester neu zu formatieren. Anschließend werden die gesicherten Dateien mit dem Dienstprogramm RESTORE wieder auf die Winchester zurückkopiert. Erst danach sind alle Änderungen erfaßt. Sie können dann wieder ordnungsgemäß arbeiten.

Fehlermeldungen

Bad sector count exceeded for this drive	Anzahl der fehlerhaften Sektoren ist zu groß für diese Platte

ERKLÄRUNG:
Es wurden mehr als 168 fehlerhafte Sektoren festgestellt. Dies könnte ein Hardwarefehler sein. Rufen Sie das Programm PREP auf. Erscheint die Fehlermeldung erneut, wenden Sie sich an Ihren Händler.

Error -- Unable to read boot code from partition	Fehler -- Es ist nicht möglich, das Boot-Byte des Teiles zu lesen

ERKLÄRUNG:
Das Boot-Byte des Teiles ist entweder nicht vorhanden oder der Sektor fehlerhaft. Booten Sie von einem anderen Laufwerk, und formatieren Sie den Teil neu. Tritt der Fehler wieder auf, muß die Platte neu initialisiert werden.

Invalid HEX value, Try again:	Falscher Hexadezimalwert, geben Sie neu ein:

ERKLÄRUNG:
Der eingegebene Wert war keine gültige hexadezimale Zahl, oder der Wert liegt außerhalb des möglichen Bereichs. Überprüfen Sie den Wert und geben Sie ihn neu ein.

Unable to communicate with the Winchester controller	Winchesterkontroller nicht vorhanden

ERKLÄRUNG:
Das Programm findet den Winchesterkontroller nicht. Es ist möglich, daß die Karte nicht richtig installiert ist. Überprüfen Sie alle Anschlüsse und starten Sie das Programm dann neu. Erscheint der Fehler wieder, ist die Karte wahrscheinlich defekt.

Teil VI

Anhang

Anhang A
Fehlermeldungen

Übersicht

Kann das *Betriebssystem* MS-DOS oder eines der *Dienstprogramme* Befehle nicht ausführen, wird eine entsprechende Fehlermeldung auf dem Bildschirm gezeigt. Die folgenden Seiten zeigen in zwei Tabellen alle Fehlertexte.

Die Tabelle A.1 ist alphabetisch nach *englischen* Texten geordnet. Zusätzlich finden Sie eine Übersetzung und als Querverweis die Seitennummer, auf der eine ausführliche Beschreibung des Fehlers mit Tips zur Beseitigung vorhanden sind.

Die Tabelle A.2 ist nach *deutschen* Fehlertexten geordnet. Sie gilt für die deutsche Version des Betriebssystemes. Auch hier ist als Querverweis die entsprechende Seitennummer vorhanden.

Fehlermeldungen

10 mismatches found

Tabelle A.1 Englische Texte

TEXT	SEITE
10 mismatches found - ending compare Es wurden mehr als 10 Fehler gefunden	**11. 69**
Acces denied Zugriff nicht möglich	**5. 30**
All data in the root directory of d will be destroyed Alle Dateien der Diskette werden gelöscht	**11. 31**
All files cancelled by operator Alle Dateien vom Bediener gelöscht	**11.224**
All files in directory will be destroyed Alle Dateien des Verzeichnisses werden gelöscht	**11. 31**
Allocate error on VM.TMP Fehler der Datei VM.TMP	**13. 20**
Allocation error, size adjusted Zuordnungsfehler, Größe korrigiert	**11. 46**
APPLY: Command string required APPLY: Befehlseingabe erforderlich	**11. 5**
APPLY: More than one command string given APPLY: Es wurden mehrere Befehle eingegeben	**11. 5**
APPLY: More than one file name specified APPLY: Es wurden mehrere Dateien benannt	**11. 5**
Attempt to access data outside of segment bounds Es wird der Versuch gemacht, auf Daten außerhalb der Segmentadresse zuzugreifen	**14. 29**
Bad call format Falscher Aufruf	**5. 30**

Tabelle A.1 Englische Texte

TEXT	SEITE
Bad command or filename Befehl oder Datei falsch	**5. 30**
Bad or missing Command Interpreter Befehlsinterpreter fehlt oder ist fehlerhaft	**9. 35**
Bad sector count exceeded for this drive Anzahl der fehlerhaften Sektoren ist zu groß für diese Platte	**17. 14**
Bad unit Falsche Einheit	**5. 31**
Bad value for cpar MaxAlloc Falscher Wert für CPARMAXALLOC	**14. 29**
Batch file missing Batchdatei fehlt	**5. 31**
BF (Bad Flag) Falsches Kennzeichen	**16. 42**
Boot sector is bad Bootsektor ist defekt	**17. 14**
BP (Too many Breakpoints) Zu viele Stopadressen	**16. 42**
BR (Bad Register) Falsches Register	**16. 42**
Cancelled by operator Datei vom Bediener gelöscht	**11.224**
Cannot access file Datei nicht vorhanden	**13. 20**

Tabelle A.1 Englische Texte

TEXT	SEITE
Cannot ASSIGN a Network drive Ein Laufwerk des Netzwerkes kann nicht umgeleitet werden	**11. 10**
Cannot CHDIR to root Das Verzeichnis kann nicht überprüft werden	**11. 47**
Cannot CHDIR to filename Das Verzeichnis kann ab Dateiname nicht weiter überprüft werden	**11. 47**
Cannot CHKDSK a Network drive Ein Laufwerk des Netzwerkes kann nicht überprüft werden	**11. 50**
Cannot create extract file Das Objektmodul kann nicht erstellt werden	**13. 20**
Cannot DISKCOMP a Network drive Ein Laufwerk des Netzwerkes kann nicht verglichen werden	**11.116**
Cannot DISKCOPY a Network drive Ein Laufwerk des Netzwerkes kann nicht kopiert werden	**11.122**
Cannot do binary reads from a device Es können keine binären Daten vom Peripheriegerät gelesen werden	**5. 31**
Cannot edit .BAK file -- rename file Datei mit dem Zusatz BAK kann nicht bearbeitet werden Datei umbenennen	**12. 47**
Cannot execute DISKCOPY Das Programm DISKCOPY kann nicht aufgerufen werden	**11.272**
Cannot execute format Das Programm FORMAT kann nicht aufgerufen werden	**11. 31**
Cannot find file Die Datei ist nicht vorhanden	**14. 29**

Tabelle A.1 Englische Texte

TEXT	SEITE
Cannot FORMAT a Network drive Ein Laufwerk des Netzwerkes kann nicht formatiert werden	**11.163**
Cannot format an assigned drive Ein umgeleitetes Laufwerk kann nicht formatiert werden	**11.165**
Cannot JOIN a Network drive Ein Laufwerk des Netzwerkes kann nicht ersetzt werden	**11.177**
Cannot LABEL a Network drive Einem Laufwerk des Netzwerkes kann kein Name gegeben werden	**11.183**
Cannot LABEL a SUBSTed or ASSIGNed drive Einem umgeleiteten Laufwerk kann kein Name gegeben werden	**11.184**
Cannot load COMMAND, system halted Die Datei COMMAND.COM kann nicht geladen werden Das System wird angehalten	**5. 32**
Cannot locate file IO.SYS Die Datei IO.SYS ist nicht vorhanden	**11. 88**
Cannot nest response file Eingabedatei darf nicht verschachtelt werden	**14. 29**
Cannot nest response file Antwortdatei kann nicht verbunden werden	**13. 20**
Cannot open response file Eingabedatei kann nicht gelesen werden	**14. 29**
Cannot open temporary file Die Arbeitsdatei kann nicht angelegt werden	**14. 30**
Cannot open VM.TMP Arbeitsdatei kann nicht angelegt werden	**13. 21**

Tabelle A.1 Englische Texte

TEXT	SEITE
Cannot read 96 Tpi on a 48 Tpi drive 96-tpi-Disketten können nicht von einem 48-tpi-Laufwerk gelesen werden	**11.237**
Cannot read partition table Die Partitiontabelle kann nicht gelesen werden	**19. 7**
Cannot read the boot sector Der Bootsektor kann nicht gelesen werden	**18. 22**
Cannot RECOVER a Network drive Ein Laufwerk des Netzwerkes kann nicht bearbeitet werden	**11.243**
Cannot SELECT a Network drive Ein Laufwerk des Netzwerkes kann nicht kopiert werden	**11.272**
Cannot start COMMAND, exiting Die Datei COMMAND.COM kann nicht aufgerufen werden Der Vorgang wird abgebrochen	**5. 32**
Cannot SUBST a Network drive Ein Verzeichnis des Netzwerkes kann nicht ersetzt werden	**11.287**
Cannot SYS a Network drive Auf ein Laufwerk des Netzwerkes können keine Systemdateien kopiert werden	**11.290**
Cannot write library file Die .LIB-Datei kann nicht geschrieben werden	**13. 21**
Cannot write the boot sector Der Bootsektor kann nicht beschrieben werden	**18. 22**
Close error on extract file Fehler in der neuen Objektdatei	**13. 21**
Configuration too large for memory Modifiziertes System für den Speicher zu groß	**9. 35**

Tabelle A.1 Englische Texte

TEXT	SEITE
Conflicting switches /F and /N specified Die Zusätze /F und /N wurden zusammen eingegeben	**11. 31**
Content of destination lost before copy Inhalt der Kette vor dem Kopieren verloren	**5. 32**
Current drive is no longer valid Aktuelles Laufwerk ist nicht mehr gültig	**5. 32**
Data error reading/writing drive X Datenfehler beim Lesen/Schreiben des Laufwerkes X	**5. 33**
DF (Double Flag) Doppeltes Kennzeichen	**16. 42**
Directory is totally empty Inhaltsverzeichnis ist leer	**11. 47**
Directory not empty Verzeichnis nicht leer	**11.177**
Disk copy failure Kopie nicht möglich	**11.122**
+++ DISK ERROR: Bad disk controller! +++ Plattenkontroller fehlerhaft	**6. 12**
+++ DISK ERROR: Cannot reset drive! +++ Laufwerk kann nicht zurückgesetzt werden	**6. 11**
+++ DISK ERROR: CRC error! +++ Fehler des CRC-Chips	**6. 12**
+++ DISK ERROR: Data corrected! +++ Daten korrigiert	**6. 11**

Fehlermeldungen

Disk not bootable

Tabelle A.1 Englische Texte

TEXT		SEITE
+++ DISK ERROR:	**Disk not bootable! +++** Diskette nicht bootfähig	6. 11
+++ DISK ERROR:	**DMA overrun error! +++** Fehler des DMA-Chips	6. 12
+++ DISK ERROR:	**Drive not ready! +++** Laufwerk nicht bereit	6. 11
+++ DISK ERROR:	**Invalid address mark detected! +++** Falsche Adresse gefunden	6. 12
+++ DISK ERROR:	**Invalid data read! +++** Fehlerhafte Daten gelesen	6. 11
+++ DISK ERROR:	**Sector not found! +++** Sektor nicht gefunden	6. 12
+++ DISK ERROR:	**Seek failure! +++** Suchfehler	6. 11
+++ DISK ERROR:	**Must run SETUP to boot from Winchester +++** Systemstart von der Winchester muß vom Programm SETUP gespeichert werden	6. 12
Disk Full -- Edits lost Diskette voll -- Datei nicht komplett gesichert		12. 47
Disk unsuitable for system disk Diskette nicht als Systemdiskette zu benutzen		11.164
Disk verify failure Diskettenprüffehler		11.115
Drive not available for CP/M reading Laufwerk kann vom Programm nicht gelesen werden		11.236

Tabelle A.1 Englische Texte

TEXT	SEITE
Drive not ready Laufwerk nicht fertig	**11.224**
Drives specified must not be the same Laufwerkangaben dürfen nicht gleich sein	**11.115**
Drives specified must not be the same Laufwerkangaben dürfen nicht gleich sein	**11.122**
Duplicate filename or File not found Dateiname doppelt oder Datei nicht gefunden	**5. 33**
Entry Error Eingabefehler	**12. 47**
Entry has a bad link Eintrag hat eine falsche Verbindung	**11. 46**
Entry has a bad size Eintrag hat eine falsche Größe	**11. 46**
EOF mark not found Das Zeichen für Dateiende ist nicht vorhanden	**11. 68**
Error - Can not communicate with the Winchester controller Winchesterkontroller nicht vorhanden	**18. 22**
Error - Disk is fully allocated to non-DOS partitions Die Platte ist komplett für Nicht-DOS-Teile eingerichtet	**18. 22**
Error - No winchester drives attached Es sind keine Winchesterlaufwerke angeschlossen	**18. 23**
Error - Requested allocation exceeds DOS maximum of 32 Mb Fehler - eingegebener Bereich ist größer als das DOS-Maximum	**18. 15**
Error - Requested allocation is less than DOS minimum Fehler - eingegebener Bereich ist kleiner als das DOS-Minimum	**18. 15**

Fehlermeldungen

Error

Tabelle A.1 Englische Texte

TEXT	SEITE
Error - Requested allocation overlaps DOS partition Fehler - eingegebener Bereich überschneidet DOS-Teil	**18. 14**
Error - Requested allocation overlaps non-DOS partition Fehler - eingegebener Bereich überschneidet Nicht-DOS-Teil	**18. 14**
Error during formatting of the drive Fehler während des Initialisierungsvorganges	**17. 14**
Error EXECing COMMAND.COM, Terminating Fehler durch Aufruf der Datei COMMAND.COM, Befehl abgebrochen	**11. 5**
Error in .EXE or .HEX file Fehler in einer .EXE- oder .-HEX-Datei	**16. 43**
Error in EXE file Fehler der EXE-Datei	**5. 33**
Error writing to cross reference file Schreibfehler der Referenzliste	**13. 21**
Error writing to device Schreibfehler des Peripheriegerätes	**5. 33**
Error writing to new library Schreibfehler der neuen LIB-Datei	**13. 21**
Error: Dup record too complex Dup-Satz zu groß	**14. 30**
Error: Fixup offset exceeds field width Fehler: Relativbestimmung ist zu groß	**14. 30**
EXE and HEX files cannot be written EXE- und .HEX-Dateien können nicht beschrieben werden	**16. 43**
EXEC failure Fehler durch Systemanruf	**5. 34**

Tabelle A.1 Englische Texte

TEXT	SEITE
EXEC failure Fehler durch Systemanruf	16. 43
Fatal Error: An internal error has occurred Schwerer Fehler: Ein interner Fehler ist aufgetreten	13. 22
Fatal Error: Cannot create list file Schwerer Fehler: Die Listdatei kann nicht erstellt werden	13. 22
Fatal Error: Cannot open response file Schwerer Fehler: Die Antwortdatei kann nicht gelesen werden	13. 22
Fatal Error: Invalid object module/library Schwerer Fehler: Objektmodul oder LIB-Datei falsch	13. 22
Fatal Error: Module is not in the library Schwerer Fehler: Objektmodul ist nicht in der .LIB-Datei	13. 22
Fatal Error: write error on library/extract file Schwerer Fehler: Schreibfehler der Dateien	13. 23
FCB unavailable FCB nicht mehr vorhanden	5. 34
File allocation table bad Die Dateizuordnungstabelle ist fehlerhaft	11. 48
File allocation table bad for drive X Dateiordnungstabelle beschädigt im Laufwerk X	5. 30
File cannot be converted Datei kann nicht umgeformt werden	15. 4
File cannot be copied onto itself Datei kann nicht auf sich selber kopiert werden	5. 34

Tabelle A.1 Englische Texte

TEXT	SEITE
File creation error Falsche Dateianlage	**5. 35**
File is READ-ONLY Datei ist schreibgeschützt	**12. 48**
File name must be specified Dateiname muß angegeben werden	**12. 48**
File not found Datei nicht gefunden	**5. 35**
Filename is cross linked on cluster x Die Datei ist ab der Gruppennummer X verbunden	**11. 48**
Files are different sizes Die Dateien haben verschiedene Größen	**11. 69**
First cluster number is invalid, entry truncated Erste Gruppennummer ist falsch, der Eintrag ist verstümmelt	**11. 47**
For cannot be nested FOR darf nicht verschachtelt werden	**5. 35**
Format failure Formatfehler	**11.164**
Format failure on destination drive Zieldiskette kann nicht formatiert werden	**11.122**
Format failure, insert another disk Fehler bei der Formatierung	**11. 32**
General failure reading/writing drive X Allgemeiner Fehler beim Lesen/Schreiben des Laufwerkes x	**5. 35**

Tabelle A.1 Englische Texte

TEXT	SEITE
Graphics characters already loaded Definition für Graphikzeichen bereits geladen	**11.169**
HARD DISK ERROR on CP/M drive Fehler des CP/M Laufwerkes	**11.236**
Has invalid cluster, file truncated Die Datei hat eine falsche Gruppennummer und ist verstümmelt	**11. 47**
I/O error Ein-/Ausgabefehler	**6. 13**
Illegal device name Falscher Name für das Peripheriegerät	**11.203**
Inalid drive/ drive not available Falsche Laufwerkbezeichnung; Bezeichnung nicht vorhanden	**19. 7**
Incompatible media Disketten nicht kompatibel	**11.115**
Incompatible system size Dateigröße verschieden	**11.291**
Input file read error Lesefehler der Datei	**14. 30**
Input file read error Lesefehler der Datei	**13. 23**
Insert disk with batch file Legen Sie die Diskette mit der Batchdatei wieder ein	**5. 31**
Insufficient disk space Nicht genug Speicherkapazität auf der Platte	**5. 35**

Tabelle A.1 Englische Texte

TEXT	SEITE
Insufficient memory Arbeitsspeicher zu klein	**15. 4**
Insufficient memory Arbeitsspeicher zu klein	**5. 36**
Insufficient memory for system transfer Der Arbeitsspeicher ist für die Systemübertragung zu klein	**11.164**
Insufficient room in root directory Nicht genug Speicherplatz im Inhaltsverzeichnis	**11. 49**
Insufficient stack space Kein Platz für das Stapelsegment vorhanden	**14. 31**
Intermediate file error during pipe Dateifehler während der Befehlsverkettung	**5. 36**
Internal table size do not match Interne Tabelle ist fehlerhaft	**11.180**
Interrupt number exceeds 255 Interrupt größer als 255	**14. 31**
Invalid backup file Fehlerhafte Datei	**11.262**
Invalid baud rate specified Falsche Baudrate eingegeben	**11.203**
Invalid characters in volume ID Falsche Zeichen im Plattennamen	**11.164**
Invalid characters in volume label Falsche Zeichen im Plattennnamen	**11.183**
Invalid COMMAND.COM Datei COMMAND.COM fehlerhaft	**5. 36**

Tabelle A.1 Englische Texte

TEXT	SEITE
Invalid country code Falsche Landeskennziffer	**11.271**
Invalid current directory Falsches Verzeichnis	**11. 49**
Invalid date Datum falsch	**5. 36**
Invalid date in switch Falsche Datumseingabe	**11. 32**
Invalid device Falsches Peripheriegerät	**5. 37**
Invalid directory Falsches Inhaltsverzeichnis	**5. 37**
Invalid disk change reading/writing drive X Nicht erlaubter Diskettentausch während Lesen/Schreiben des Laufwerkes X	**5. 37**
Invalid drive designation on RESTORE file Falsche Laufwerkbezeichnung	**11.262**
Invalid drive in search path Falsche Laufwerkbezeichnung im Weg	**5. 37**
Invalid drive or file name Falsches Laufwerk oder Dateiname	**12. 49**
Invalid drive specification Falsche Laufwerkbezeichnung	**5. 37**
Invalid exception file specifications Falsche Dateibezeichnung für die mit /E bezeichneten Dateien	**11.32**

Invalid filename

Tabelle A.1 Englische Texte

TEXT	SEITE
Invalid filename or file not found Falscher Dateiname oder Datei nicht vorhanden	**5. 37**
Invalid keyboard code Falscher Name für Tastaturprogramm	**11.272**
Invalid number of parameters Falsche Anzahl von Zusatzeingaben	**5. 38**
Invalid numeric switch specification Falscher Zahlenwert	**14. 31**
Invalid object module Fehlerhaftes Modul	**14. 31**
Invalid parameter Falsche Zusatzeingabe	**5. 38**
Invalid partition selection Falsche Bezeichnung des Bereiches	**19. 7**
Invalid path or file name Weg oder Dateiname falsch	**5. 38**
Invalid path, not directory, or directory not empty Falscher Weg, kein Verzeichnis oder Verzeichnis nicht leer	**5. 38**
Invalid selection file specifications Falsche Eingabe der Dateiauswahl	**11. 33**
Invalid subdirectory entry Falscher Eintrag im Verzeichnis	**11. 50**
Invalid switch /X specified. Falsche Zusatzeingabe /X benannt	**11. 33**

Tabelle A.1 Englische Texte

TEXT	SEITE
Invalid time Falsche Zeit	**5. 38**
Invalid value for /X ignored Falsche Zusatzeingabe /X nicht beachtet	**11.224**
Invalid version of IO.SYS Falsche Version der Datei IO.SYS	**19. 8**
Invalid Winchester drive number given Falsche Nummer des Winchesterlaufwerkes eingegeben	**19. 8**
IO.SYS file version incorrect Die Version der Datei IO.SYS ist nicht kompatibel zum Programm	**11. 88**
Label not found Bezeichnung nicht vorhanden	**5. 39**
Library Disk is full Diskette ist voll	**13. 23**
Line too long Zeile zu lang	**12. 49**
List output not assigned to a device Eingegebener Name bezeichnet kein Ausgabegerät	**11.225**
Listing file write error Schreibfehler der Listdatei	**13. 24**
Lock-Violation reading/writing drive X Datensatz gesperrt beim Lesen/Schreiben des Laufwerkes X	**5. 39**
Memory allocation error Fehler bei Speicherzuweisung	**5. 39**
Missing country code Landeskennziffer fehlt	**11.272**

Missing keyboard code

Tabelle A.1 Englische Texte

TEXT	SEITE
Missing keyboard code Name für Tastaturprogramm fehlt	**11.272**
Must specify destination line number Geben Sie eine Zeilennummer ein	**12. 49**
Nested right parenthesis Rechte Klammer zu viel	**14. 31**
No bootable partitions Keine bootfähigen Teile	**6. 13**
No files match Keine Datei entspricht	**11.225**
No files selected Keine Dateien benannt	**11. 33**
No free file handles Keine Dateinummer mehr frei	**5. 39**
No library file specified Keine .LIB-Datei benannt	**13. 24**
No object modules specified Keine Objektmodule benannt	**14. 32**
No paper writing device Kein Papier, Fehler beim Schreiben des Peripheriegerätes	**5. 40**
No room for system on destination disk Kein Speicherplatz für die Systemdateien auf der Zieldiskette	**11.291**
No room in directory for file Kein Platz im Verzeichnis für die Datei	**12. 49**
No room in directory to create file Kein Platz mehr im Inhaltsverzeichnis	**11.237**

Tabelle A.1 Englische Texte

TEXT	SEITE
No room in root directory Kein Platz im Rootdir	**11.184**
No serial or parallel devices are attached Es sind keine seriellen oder parallelen Ausgänge angeschlossen	**11. 88**
No System Kein System	**6. 13**
Non-DOS disk reading/writing drive X Keine DOS-Diskette beim Lesen/Schreiben des Laufwerkes X	**5. 39**
Not a bootable partition Kein bootfähiger Teil	**6. 13**
Not a DOS partition Kein DOS-Teil	**19. 8**
Not able to back up this time Die Datei kann momentan nicht gesichert werden	**11. 33**
Not enough memory Arbeitsspeicher zu klein	**11.177**
Not enough parameters specified Nicht alle Eingaben vorgenommen	**11. 34**
Not enough room to merge the entire file Arbeitsspeicher zu klein Die gesamte Datei kann nicht geladen werden	**12. 50**
Not ready drive X Laufwerk X nicht bereit	**5. 39**
Out of disk on restoration of . . . Speicherkapazität der Platte ist für die Datei . . . erschöpft	**11.264**

Tabelle A.1 Englische Texte

TEXT	SEITE
Out of environment space Speicher für die Systemvariablen belegt	**5. 40**
Output library specification ignored Name für die neue LIB-Datei nicht beachtet	**13. 24**
Page size too small Seitengröße zu klein	**13. 24**
Parity error or nonexistant memory error detected Fehler des Arbeitsspeichers gefunden	**16. 44**
Partition is already in use Der Bereich ist bereits benannt	**19. 8**
Partition is too large Der Bereich ist zu groß	**19. 9**
Partition is too small Der Bereich ist zu klein	**19. 9**
Partition not formatted Winchesterteil nicht formatiert	**6. 13**
Probable non-DOS disk Keine DOS-Diskette	**11. 50**
Program size or number of segments exceeds capacity of Linker Programmgröße oder Anzahl der Segmente übersteigt die Kapazität von LINK	**14. 32**
Programm too big to fit in memory Das Programm ist für den Speicher zu groß	**5. 40**
Read error in: Lesefehler der Datei:	**12. 50**

Tabelle A.1 Englische Texte

TEXT	SEITE
Read error on destination drive Lesefehler der Zieldiskette	**11.123**
Read error on Source drive Lesefehler der Ursprungsdiskette	**11.123**
Read error on Source1 drive Lesefehler auf der ersten Diskette	**11.116**
Read error on Source2 drive Lesefehler auf der zweiten Diskette	**11.116**
Read error on VM.TMP Lesefehler der Datei VM.TMP	**13. 24**
Read fault drive X Lesefehler des Laufwerkes X	**5. 40**
Sector not found Sektor nicht gefunden	**5. 41**
Sector size too large in file Sektor zu groß in Datei	**9. 35**
Seek error reading/writing drive X Suchfehler beim Lesen/Schreiben des Laufwerkes X	**5. 41**
Segment limit set too high Segmentgrenze zu hoch	**14. 32**
Segment size exceeds 64Kb Segment ist größer als 64Kb	**14. 32**
SHARE already installed Das Programm SHARE wurde bereits aufgerufen	**11.278**

Tabelle A.1 Englische Texte

TEXT	SEITE
Sharing violation reading/writing drive X Datei gesperrt beim Lesen/Schreiben des Laufwerkes X	**5. 41**
Source and destination drive must not be the same Ursprungs- und Ziellaufwerk dürfen nicht gleich sein	**11.236**
Source file not found Datei nicht gefunden	**11.237**
Stack size exceeds 64Kb Stapelsegment ist größer als 64Kb	**14. 32**
Symbol already defined Symbol mehr als einmal definiert.	**14. 32**
Symbol is a multiply defined PUBLIC. Proceed? Das Symbol ist mehrfach definiert. Fortfahren?	**13. 20**
Symbol table capacity exceeded Speicherkapazität der Symboltabelle überschritten	**14. 33**
Syntax error (bad file spec) Schreibfehler (Falsche Dateibezeichnung)	**13. 25**
Too many Block Devices Zu viele Gerätetreiber bestimmt	**9. 35**
Too many external symbols in one module Zu viele externe Symbole in einem Modul	**14. 33**
Too many files open Keine Dateinummer mehr frei	**11.184**
Too many groups Zu viele Gruppen	**14. 33**
Too many libraries specified Zu viele Bibliotheken benannt	**14. 33**

Tabelle A.1 Englische Texte

TEXT	SEITE
Too many objekt modules Zu viele Objektmodule	**13. 25**
Too many overlays Zu viele Overlaymodule	**14. 33**
Too many parameters specified Zu viele Zusatzeingaben	**11. 34**
Too many PUBLIC symbols Zu viele PUBLIC-Symbole	**14. 33**
Too many segments or classes Zu viele Segmente oder Klassen	**14. 34**
Too many subdirectories Zu viele Verzeichnisse	**11. 34**
Track 0 bad - disk unusable Spur 0 beschädigt - Diskette ist nicht zu gebrauchen	**11.165**
Unable to communicate with the Winchester controller Winchesterkontroller nicht vorhanden	**17. 14**
Unable to create directory Verzeichnis kann nicht angelegt werden	**5. 41**
Unable to load Key map routine Programm kann nicht geladen werden	**11.180**
Unable to read boot code from partition Das Bootbyte kann vom Laufwerk nicht gelesen werden	**6. 13**
Unknown switch Falscher Zusatz	**13. 25**
Unrecognized command in CONFIG.SYS Nicht erkannter Befehl in der Datei CONFIG.SYS	**9. 36**

Unrecoverable error

Tabelle A.1 Englische Texte

TEXT	SEITE
Unrecoverable error in directory Nicht behebbarer Fehler im Verzeichnis	11. 50
Unresolved externals: Ungelöste Querreferenzen	14. 34
Version mismatch with IO.SYS Falsche Version der Datei IO.SYS	11. 88
VM read error Lesefehler der Datei VM.TMP	14. 34
Warning - directory full Achtung - Inhaltsverzeichnis voll	11.243
WARNING - Read error on EXE file WARNUNG - Lesefehler der EXE Datei	15. 5
Warning: No STACK segment Warnung: Kein Stapelsegment	14. 34
Write error on destination drive Schreibfehler der Zieldiskette	11.123
Write error on VM.TMP Schreibfehler der Datei VM.TMP	13. 25
Write fault Schreibfehler des Laufwerkes X	5. 42
Write protect writing drive X Schreibschutzfehler beim Schreiben des Laufwerkes X	5. 42

Tabelle A.2 Deutsche Texte

TEXT	SEITE
96-tpi-Disketten können nicht von einem 48-tpi-Laufwerk gelesen werden	11.237
Achtung - Inhaltsverzeichnis voll	11.243
Aktuelles Laufwerk ist nicht mehr gültig	5. 32
Alle Dateien der Diskette werden gelöscht	11. 31
Alle Dateien des Verzeichnisses werden gelöscht	11. 31
Alle Dateien vom Bediener gelöscht	11.224
Allgemeiner Fehler beim Lesen/Schreiben des Laufwerkes X	5. 35
Antwortdatei kann nicht verbunden werden	13. 20
Anzahl der fehlerhaften Sektoren ist zu groß für diese Platte	17. 14
APPLY: Befehlseingabe erforderlich	11. 5
APPLY: Es wurden mehrere Befehle eingegeben	11. 5
APPLY: Es wurden mehrere Dateien benannt	11. 5
Arbeitsspeicher zu klein	5. 36
Auf ein Laufwerk des Netzwerkes können keine Systemdateien kopiert werden	11.290
Batchdatei fehlt	5. 31
Befehl oder Datei falsch	5. 30
Befehlsinterpreter fehlt oder ist fehlerhaft	9. 35

Bezeichnung nicht vorhanden

Tabelle A.2 Deutsche Texte

TEXT	SEITE
Bezeichnung nicht vorhanden	5. 39
Bootsektor ist defekt	17. 14
Das Objektmodul kann nicht erstellt werden	13. 20
Das Programm DISKCOPY kann nicht aufgerufen werden	11.272
Das Programm FORMAT kann nicht aufgerufen werden	11. 31
Das Programm ist für den Speicher zu groß	5. 40
Das Programm SHARE wurde bereits aufgerufen	11.278
Das Symbol ist mehrfach definiert. Fortfahren?	13. 20
Das Verzeichnis kann nicht überprüft werden	11. 47
Das Verzeichnis kann ab Dateiname nicht weiter überprüft werden	11. 47
Das Zeichen für Dateiende ist nicht vorhanden.	11. 68
Datei COMMAND.COM fehlerhaft	5. 36
Datei gesperrt beim Lesen/Schreiben des Laufwerkes X	5. 41
Datei ist schreibgeschützt	12. 48
Datei kann nicht auf sich selber kopiert werden	5. 34
Datei kann nicht umgeformt werden	15. 4
Datei mit dem Zusatz .BAK kann nicht bearbeitet werden	
Datei umbenennen	12. 47
Datei nicht gefunden	11.237

Tabelle A.2 Deutsche Texte

TEXT	SEITE
Datei nicht vorhanden	13. 20
Datei vom Bediener gelöscht	11.224
Dateifehler während der Befehlsverkettung	5. 36
Dateigröße verschieden	11.291
Dateiname doppelt oder Datei nicht gefunden	5. 33
Dateiname muß angegeben werden	12. 48
Dateiordnungstabelle beschädigt im Laufwerk X	5. 30
Datenfehler beim Lesen/Schreiben des Laufwerkes X	5. 33
Datensatz gesperrt beim Lesen/Schreiben des Laufwerkes X	5. 39
Datum falsch	5. 36
Definition für Graphikzeichen bereits geladen	11.169
Fehler des CP/M Laufwerkes	11.236
Der Arbeitsspeicher ist für die Systemübertragung zu klein	11.164
Der Bereich ist bereits benannt	19. 8
Der Bereich ist zu groß	19. 9
Der Bereich ist zu klein	19. 9
Das Bootbyte kann vom Laufwerk nicht gelesen werden	6. 13
Der Bootsektor kann nicht beschrieben werden	18. 22
Der Bootsektor kann nicht gelesen werden	18. 22

Fehlermeldungen

Die LIB-Datei

Tabelle A.2 Deutsche Texte

TEXT	SEITE
Die .LIB-Datei kann nicht geschrieben werden	13. 21
Die Arbeitsdatei kann nicht angelegt werden	14. 30
Die Datei COMMAND.COM kann nicht aufgerufen werden	5. 32
Die Datei COMMAND.COM kann nicht geladen werden Das System wird angehalten	5. 32
Die Datei hat eine falsche Gruppennummer und ist verstümmelt	11. 47
Die Datei IO.SYS ist nicht vorhanden	11. 88
Die Datei ist ab der Gruppennummer X verbunden	11. 48
Die Datei ist nicht vorhanden	14. 29
Die Datei kann momentan nicht gesichert werden	11. 33
Die Dateien haben verschiedene Größen	11. 69
Die Dateizuordnungstabelle ist fehlerhaft	11. 48
Die Partition-Tabelle kann nicht gelesen werden	19. 7
Die Platte ist komplett für Nicht-DOS-Teile eingerichtet	18. 22
Die Version der Datei IO.SYS ist nicht kompatibel zum Programm	11. 88
Die Zusätze /F und /N wurden zusammen eingegeben	11. 31
Inhalt der Kette vor dem Kopieren verloren	5. 32
Diskette ist voll	13. 23
Diskette nicht als Systemdiskette zu benutzen	11.164
Diskette voll -- Datei nicht komplett gesichert	12. 47

Tabelle A.2 Deutsche Texte

TEXT	SEITE
Disketten nicht kompatibel	11.115
Diskettenprüffehler	11.115
Doppeltes Kennzeichen	16. 42
Dup-Satz zu groß	14. 38
Ein Laufwerk des Netzwerkes kann nicht überprüft werden	11. 50
Ein Laufwerk des Netzwerkes kann nicht bearbeitet werden	11.243
Ein Laufwerk des Netzwerkes kann nicht ersetzt werden	11.177
Ein Laufwerk des Netzwerkes kann nicht formatiert werden	11.163
Ein Laufwerk des Netzwerkes kann nicht kopiert werden	11.122
Ein Laufwerk des Netzwerkes kann nicht umgeleitet werden	11. 10
Ein Laufwerk des Netzwerkes kann nicht verglichen werden	11.116
Ein umgeleitetes Laufwerk kann nicht formatiert werden	11.165
Ein Verzeichnis des Netzwerkes kann nicht ersetzt werden	11.287
Ein-/Ausgabefehler	6. 13
Einem Laufwerk des Netzwerkes kann kein Name gegeben werden	11.183
Einem umgeleiteten Laufwerk kann kein Name gegeben werden	11.184
Eingabedatei darf nicht verschachtelt werden	14. 29
Eingabedatei kann nicht gelesen werden	14. 29
Eingabefehler	12. 47

Tabelle A.2 Deutsche Texte

TEXT	SEITE
Eingegebener Name bezeichnet kein Ausgabegerät	11.225
Eintrag hat eine falsche Größe	11. 46
Eintrag hat eine falsche Verbindung	11. 46
Erste Gruppennummer ist falsch, der Eintrag ist verstümmelt	11. 47
Es können keine binären Daten vom Peripheriegerät gelesen werden	5. 31
Es sind keine seriellen oder parallelen Ausgänge angeschlossen	11. 88
Es sind keine Winchesterlaufwerke angeschlossen	18. 23
Es wurden mehr als 10 Fehler gefunden	11. 69
EXE- und .HEX-Dateien können nicht beschrieben werden	16. 43
Es wird der Versuch gemacht, auf Daten außerhalb der Segmentadresse zuzugreifen	14. 29
Falsche Anzahl von Zusatzeingaben	5. 38
Falsche Baudrate eingegeben	11.203
Falsche Bezeichnung des Bereiches	19. 7
Falsche Dateianlage	5. 35
Falsche Dateibezeichnung für die mit /E bezeichneten Dateien	11. 32
Falsche Datumseingabe	11. 32
Falsche Eingabe der Dateiauswahl	11. 33
Falsche Einheit	5. 31
Falsche Landeskennziffer	11.272

Tabelle A.2 Deutsche Texte

TEXT	SEITE
Falsche Laufwerkbezeichnung	11.262
Falsche Laufwerkbezeichnung im Weg	5. 37
Falsche Laufwerkbezeichnung; Bezeichnung nicht vorhanden	19. 7
Falsche Nummer des Winchesterlaufwerkes eingegeben	19. 8
Falsche Version der Datei IO.SYS	11. 88
Falsche Version der Datei IO.SYS	19. 8
Falsche Zeichen im Plattennamen	11.164
Falsche Zeit	5. 38
Falsche Zusatzeingabe /X benannt	14. 33
Falsche Zusatzeingabe /X nicht beachtet	11.224
Falsche Zusatzeingabe	5. 38
Falscher Aufruf	5. 30
Falscher Dateiname oder Datei nicht vorhanden	5. 37
Falscher Eintrag im Verzeichnis	11. 50
Falscher Name für das Peripheriegerät	11.203
Falscher Name für Tastaturprogramm	11.272
Falscher Weg, kein Verzeichnis oder Verzeichnis nicht leer	5. 38
Falscher Wert für CPARMAXALLOC	14. 29
Falscher Zahlenwert	14. 31

Fehlermeldungen

Falscher Zusatz

Tabelle A.2 Deutsche Texte

TEXT	SEITE
Falscher Zusatz	13. 25
Falsches Inhaltsverzeichnis	5. 37
Falsches Kennzeichen	16. 42
Falsches Laufwerk oder Dateiname	12. 49
Falsches Peripheriegerät	5. 37
Falsches Register	16. 42
Falsches Verzeichnis	11. 49
FCB nicht mehr vorhanden	5. 34
Fehler - eingegebener Bereich überschneidet DOS-Teil	18. 14
Fehler - eingegebener Bereich überschneidet Nicht-DOS-Teil	18. 14
Fehler - eingegebener Bereich ist größer als das DOS-Maximum	18. 15
Fehler - eingegebener Bereich ist kleiner als das DOS-Minimum	18. 15
Fehler bei Speicherzuweisung	5. 39
Fehler der Datei VM.TMP	13. 20
Fehler der EXE-Datei	5. 33
Fehler des Arbeitsspeichers gefunden	16. 44
Fehler durch Aufruf der Datei COMMAND.COM, Befehl abgebrochen	11. 5
Fehler durch Systemanruf	5. 34
Fehler in der neuen Objektdatei	13. 21

Tabelle A.2 Deutsche Texte

TEXT	SEITE
Fehler in einer .EXE- oder .-HEX-Datei	16. 43
Fehler während des Initialisierungsvorganges	17. 14
Fehler: Relativbestimmung ist zu groß	14. 38
Fehlerhafte Datei	11.262
Fehlerhaftes Modul	14. 31
Fehler bei der Formatierung	11. 32
FOR darf nicht verschachtelt werden	5. 35
Formatfehler	11.164
Geben Sie eine Zeilennummer ein	12. 49
Inhaltsverzeichnis ist leer	11. 47
Interne Tabelle ist fehlerhaft	11.180
Interrupt größer als 255	14. 31
Kein bootfähiger Teil	6. 13
Kein DOS-Teil	19. 8
Kein Papier, Fehler beim Schreiben des Peripheriegerätes	5. 40
Kein Platz für das Stapelsegment vorhanden	14. 31
Kein Platz im Rootdir	11.184
Kein Platz im Verzeichnis für die Datei	12. 49
Kein Platz mehr im Inhaltsverzeichnis	11.237

Kein Speicherplatz

Tabelle A.2 Deutsche Texte

TEXT	SEITE
Kein Speicherplatz für die Systemdateien auf der Zieldiskette	11.291
Kein System	6. 13
Keine .LIB-Datei benannt	13. 24
Keine bootfähigen Teilbereiche	6. 13
Keine Datei entspricht	11.225
Keine Dateien benannt	11. 33
Keine Dateinummer mehr frei	11.184
Keine DOS-Diskette	11. 50
Keine DOS-Diskette beim Lesen/Schreiben des Laufwerkes X	5. 39
Keine Objektmodule benannt	14. 32
Kopie nicht möglich	11.122
Landeskennziffer fehlt	11.272
Laufwerk kann vom Programm nicht gelesen werden	11.236
Laufwerk nicht fertig	11.224
Laufwerk X nicht bereit	5. 39
Laufwerkangaben dürfen nicht gleich sein	11.115
Legen Sie die Diskette mit der Batchdatei wieder ein	5. 31
Lesefehler auf der ersten Diskette	11.116
Lesefehler auf der zweiten Diskette	11.116

Tabelle A.2 Deutsche Texte

TEXT	SEITE
Lesefehler der Datei	14. 30
Lesefehler der Datei VM.TMP	13. 24
Lesefehler der Datei VM.TMP	14. 34
Lesefehler der Datei:	12. 50
Lesefehler der Ursprungsdiskette	11.123
Lesefehler der Zieldiskette	11.123
Lesefehler des Laufwerkes X	5. 40
Modifiziertes System für den Speicher zu groß	9. 35
Name für das Tastaturprogramm fehlt	11.272
Name für die neue LIB-Datei nicht beachtet	13. 24
Nicht alle Eingaben vorgenommen	11. 34
Nicht erkannter Befehl in der Datei CONFIG.SYS	9. 36
Nicht erlaubter Diskettentausch	5. 37
Nicht genug Speicherkapazität auf der Platte	5. 35
Nicht genug Speicherplatz im Inhaltsverzeichnis	11. 49
Nicht behebbarer Fehler im Verzeichnis	11. 50
Programm kann nicht geladen werden	11.180
Programmgröße oder Anzahl der Segmente übersteigt die Kapazität von LINK	14. 32
Rechte Klammer zu viel	14. 31

Fehlermeldungen

Schreibfehler

Tabelle A.2 Deutsche Texte

TEXT	SEITE
Schreibfehler (Falsche Dateibezeichnung)	13. 25
Schreibfehler der Datei VM.TMP	13. 25
Schreibfehler der Listdatei	13. 24
Schreibfehler der neuen LIB-Datei	13. 21
Schreibfehler der Referenzliste	13. 21
Schreibfehler der Zieldiskette	11.123
Schreibfehler des Laufwerkes X	5. 42
Schreibfehler des Peripheriegerätes	5. 33
Schreibschutzfehler beim Schreiben des Laufwerkes X	5. 42
Schwerer Fehler: Die Antwortdatei kann nicht gelesen werden	13. 22
Schwerer Fehler: Die Listdatei kann nicht erstellt werden	13. 22
Schwerer Fehler: Ein interner Fehler ist aufgetreten	13. 22
Schwerer Fehler: Objektmodul ist nicht in der .LIB-Datei	13. 22
Schwerer Fehler: Objektmodul oder LIB-Datei falsch	13. 22
Schwerer Fehler: Schreibfehler der Dateien	13. 23
Segment ist größer als 64Kb	14. 32
Segmentgrenze zu hoch	14. 32
Seitengröße zu klein	13. 24
Sektor nicht gefunden	5. 41

Tabelle A.2 Deutsche Texte

TEXT	SEITE
Sektor zu groß in Datei	9. 35
Speicher für die Systemvariablen belegt	5. 40
Speicherkapazität der Platte ist für die Datei . . . erschöpft	11.264
Speicherkapazität der Symboltabelle überschritten	14. 33
Spur 0 beschädigt - Diskette ist nicht zu gebrauchen	11.165
Stapelsegment ist größer als 64Kb	14. 32
Suchfehler beim Lesen/Schreiben des Laufwerkes X	5. 41
Symbol mehr als einmal definiert	14. 32
Ungelöste Querreferenzen	14. 34
Ursprungs- und Ziellaufwerk dürfen nicht gleich sein	11.236
Verzeichnis kann nicht angelegt werden	5. 41
Verzeichnis nicht leer	11.177
WARNUNG - Lesefehler der EXE-Datei.	15. 5
Warnung: Kein Stapelsegment	14. 34
Weg oder Dateiname falsch	5. 38
Winchesterkontroller nicht vorhanden	17. 14
Winchesterkontroller nicht vorhanden	18. 22
Winchesterteil nicht formatiert	6. 13
Zeile zu lang	12. 49
Zieldiskette kann nicht formatiert werden	11.122

Zu viele Bibliotheken

Tabelle A.2 Deutsche Texte

TEXT	SEITE
Zu viele Bibliotheken benannt	14. 33
Zu viele externe Symbole in einem Modul	14. 33
Zu viele Gerätetreiber bestimmt	9. 35
Zu viele Gruppen	14. 33
Zu viele Objektmodule	13. 25
Zu viele Overlaymodule	14. 33
Zu viele Segmente oder Klassen	14. 34
Zu viele Stopadressen	16. 42
Zu viele Verzeichnisse	11. 34
Zu viele Zusatzeingaben	11. 34
Zu viele PUBLIC-Symbole	14. 33
Zugriff nicht möglich	5. 30
Zuordnungsfehler, Größe korrigiert	11. 46

Anhang B

Programme der deutschen Version

Übersicht

Das vorliegende Handbuch für das Betriebssystem MS-DOS 3.10 wurde hauptsächlich für die englische Originalfassung geschrieben. Es beschreibt alle englischen Programme mit ihren Funktionen. Zusätzlich haben Sie jedoch auch eine übersetzte Version des Betriebssystemes erhalten. Eine Beschreibung der ins deutsche übersetzten sowie der zusätzlichen Programme erfolgt in diesem gesonderten Kapitel.

Die deutsche Version des Betriebssystemes unterscheidet sich von der Originalfassung durch:

- Zusatzfunktionen für alle Programme
- zusätzliche Befehle für bestimmte Programme
- zusätzliche Programme

Eine ausführliche Beschreibung finden Sie auf den folgenden Seiten.

Allgemeine Unterschiede

Alle Programme der *deutschen* Version löschen nach Ihrem Start zuerst den Bildschirm und zeigen dann den Programmnamen mit der Versionsnummer. Gleichzeitig erfolgt eine Prüfung des Betriebssystemes. Alle Programme arbeiten erst ab der Versionsnummer 3.10 fehlerfrei. Benutzen Sie die Dienstprogramme unter einer niedrigeren Versionsnummer, erscheint die Fehlermeldung

```
Falsche DOS-Version
```

und das Programm kehrt zur Systemebene zurück. Benutzen Sie die Dienstprogramme auf einem kompatiblen Personalcomputer, können Sie ebenfalls nicht fehlerfrei arbeiten. Es erscheint dann die Meldung

```
Nicht lizensierte Kopie
```

und das Programm kehrt ebenfalls zur Systemebene zurück.

Jedes Dienstprogramm enthält Informationen zur Arbeitsweise und zur Befehlseingabe. Rufen Sie diese Hilfen durch Eingabe eines Fragezeichens auf. Geben Sie z. B. ein

```
PRINT ?
```

erscheint eine kurze Beschreibung des Programmes PRINT (s. a. Seite 11.213) mit allen Zusatzeingaben. Diese Möglichkeit besteht für alle direkt aufrufbaren Programme, bis auf die Datei COMMAND.COM.

Benutzen Sie die Filterprogramme, erscheint ebenfalls der Programmname mit Versionsnummer. Der Text wird dabei immer auf den Bildschirm ausgegeben und ist nicht umleitbar. Damit können bei Datenumleitung alle Daten des Filters wie gewohnt weiterbearbeitet werden. Verketten Sie Filter mit weiteren Programmen, wird der Bildschirm vom Filter nicht mehr gelöscht. Ebenso erfolgt dann keine Anzeige des Programmnamens.

Zusatzfunktionen

Außer den allgemeinen Unterschieden verfügen drei Programme über zusätzliche Befehle.

COMMAND.COM

Die Datei COMMAND.COM ist der Interpreter des Betriebssystemes und enthält alle internen Befehle. Daher kann sie nicht mit einem einzelnen Fragezeichen aufgerufen werden. Sie verfügt aber ebenfalls über Hilfsinformationen. Diese werden über einen neuen internen Befehl aufgerufen. Geben Sie ein:

HILFE

und es erscheint folgende Übersicht:

```
MS-DOS Version 3.10D
Copyright (C) 1985, Zenith Data Systems Corporation

>>> INTERNE BEFEHLE <<<

BREAK - [ON] [OFF]                      MKDIR  - [d:] [Weg]
CHDIR - [d:] [Weg]                      PATH   - [d:] [\Verzeichnis[\...]]
CLS   -                                 PAUSE  - [Text]
COPY  - Dateibez Dateiname [/X]         PROMPT - [$] [Text]
CTTY  - Device                          REM    - [Text]
DATE  - [tt-mm-jj]                      REN    - Dateibez Dateiname
DEL   - Dateibez                        RMDIR  - [d:] [Weg]
DIR   - [Dateibez] [/P] [/W]            SET    - [Zeichenkette1 =
                                                 [Zeichenkette2]]
ECHO  - [Text] [ON] [OFF]               SHIFT  -
EXIT  -                                 TIME   - [hh[:mm[:ss[.cc]]]]
FOR   - %Variable IN (Wert) DO Befehl   TYPE   - Dateibez
GOTO  - Bezeichnung                     VER    -
HILFE - es erscheint diese Übersicht    VERIFY - [ON] [OFF]
IF    - [NOT] Bedingung Befehl          VOL    - [d:]
MKDIR - [d:] [Weg]
```

Zusatzfunktionen

DEBUG.COM

Außer den Befehlen und Funktionen, die schon im Kapitel 16 beschrieben wurden, enthält die deutsche Version des Programmes drei neue Befehle und eine neue Funktion. Außerdem ist auch hier die Eingabe eines einzelnen Fragezeichens möglich.

Anzeige der Register

Nach dem Programmstart wird der Bildschirm gelöscht, es erscheint der Programmname mit Versionsnummer, und direkt darunter der aktuelle Inhalt aller Register. Dies entspricht der Eingabe des Befehles **R**(egister). Das Programm meldet sich mit folgendem Bild:

```
                    DEBUG Version 2.30D
       Copyright (C) 1985, Zenith Data Systems Corporation

AX=0000 BX=0000 CX=0000 DX=0000 SP=C4DE BP=0000 SI=0000 DI=0000
DS=10DA ES=10DA SS=10DA CS=10DA IP=0100 NV UP EI PL NZ NA PO NC
10DA:0100 56            PUSH    SI
```

B -- Befehlsübersicht

Mit Eingabe eines einzelnen B rufen Sie jederzeit eine Übersicht aller verfügbaren Befehle mit möglichen Zusatzeingaben auf. Geben Sie ein:

```
B
```

erscheint folgende Übersicht:

DEBUG - Befehlsübersicht
Copyright (C) 1985, Zenith Data Systems Corporation

Assemble	A [Adresse]
Befehlsübersicht	B
Compare	C Bereich Adresse
Dump	D [Bereich] [Adresse] [L Wert]
Enter	E Adresse [Liste]
Fill	F Bereich Liste
Go	G [=Adresse1] [Adresse2...]
Hex	H Wert Wert
Input	I Wert
Kommentar	K
Load	L [Adresse [Laufwerk Sektor Anzahl]]
Move	M Bereich Adresse
Name	N Dateibez [Dateibez...]
Output	O Wert Byte
Proceed	P [=[Adresse] [Wert]]
Quit	Q
Register	R [Register]
Search	S Bereich Liste
Trace	T [=[Adresse] [Wert]]
Unassemble	U [Bereich] [Adresse] [L Wert]
Write	W [Adresse [Laufwerk Sektor Anzahl]]
ZLS	Z

K -- Kommentar

Das Programm DEBUG liest die Standardeingabe des Betriebssystemes. Dies ist normalerweise die Tastatur. Mit Hilfe des Zeichens „<“ kann es aber auch eine Datei sein. Dann erhält das Programm alle Befehle von der Datei. Eine Eingabe über die Tastatur ist nicht mehr möglich. Achten Sie daher immer darauf, eine Eingabedatei mit dem Befehl **Q**(uit) abzuschließen.

Werden umfangreichere Änderungen eines Programmes mit Hilfe einer Eingabedatei vorgenommen, sollten alle Eingaben kommentiert werden. Nur so kann die Datei für spätere Änderungen wieder benutzt werden. Hierfür eignet sich der neue Befehl **K**(ommentar) sehr gut. Alle Zeichen nach dem Befehlsbuchstaben werden nicht beachtet. Es könnte z. B. folgender Kommentar benutzt werden:

K Eingabedatei für DEBUG.COM, Stand: 15.11.1985

Zusatzfunktionen

Z -- Bildschirm löschen

Mit diesem Befehl kann jederzeit der Bildschirm gelöscht werden. Gleichzeitig wird der Cursor in die linke obere Ecke positioniert.

EDLIN.EXE

Außer den Befehlen und Funktionen, die schon im Kapitel 12 beschrieben wurden, enthält die deutsche Version des Programmes einen neuen Befehl. Außerdem ist auch hier die Eingabe eines einzelnen Fragezeichens möglich.

B -- Befehlsübersicht

Mit Eingabe eines einzelnen B rufen Sie jederzeit eine Übersicht aller verfügbaren Befehle mit möglichen Zusatzeingaben auf. Geben Sie ein

B

erscheint folgende Übersicht:

EDLIN - Befehlsübersicht
Copyright (C) 1985, Zenith Data Systems Corporation

BEFEHL	EINGABE
Append Lines	[Anzahl] A
Befehlsübersicht	B
Copy Lines	[ZeileA], [ZeileE], ZeileN, [Anzahl] C
Delete Lines	[ZeileA] [,ZeileE] D
Edit Line	Zeile
End Edit	E
Insert Line	Zeile I
List Lines	[ZeileA] [,ZeileE] L
Move Lines	[ZeileA], [ZeileE], ZeileN M
Page Text	[ZeileA], [ZeileE] P
Quit Edit	Q
Replace Text	[ZeileA] [,ZeileE] [?] R [Zeichenkette1] F6 [Zeichenkette2]
Search Line	[ZeileA] [,ZeileE] [?] S [Zeichenkette]
Transfer Lines	[ZeileN] T Dateiname
Write Lines	[Anzahl] W

ZDIR Extern

Zweck

zeigt alle Einträge des Inhaltsverzeichnisses mit allen gespeicherten Informationen

Syntax

[d:] [Weg] ZDIR ?
[d:] [Weg] ZDIR [d:] Dateiname
[d:] [Weg] ZDIR [d:] Weg

dabei bedeutet:

d: vor dem Programmnamen bestimmt das aufzurufende Laufwerk

Weg vor dem Programmnamen bestimmt das aufzurufende Verzeichnis

? es erscheint eine Übersicht des Programmes

d: nach dem Programmnamen bezeichnet das Laufwerk, dessen Inhaltsverzeichnis gezeigt werden soll

Dateiname Name(n) der Einträge, deren Informationen gezeigt werden sollen

Weg nach dem Programmnamen bezeichnet mehrere Verzeichnisse in Folge in der Form:

[\] [Verzeichnis [Verzeichnis...]]

ZDIR

Bemerkungen

Das Programm ZDIR kann im *Netzwerk* **ohne** Einschränkung benutzt werden.

Das Programm ZDIR ist eine Ergänzung des internen Befehles DIR. Mit diesem Programm werden alle Einträge des Inhaltsverzeichnisses (Dateien, Unterverzeichnisse und Plattenname) mit allen gespeicherten Informationen angezeigt. Zusätzlich erscheinen im oberen Teil des Bildschirmes allgemeine Informationen über das Laufwerk, sowie Datum und Uhrzeit. Es werden maximal 18 Einträge gezeigt, dann stoppt die Ausgabe. Nach Betätigen einer beliebigen Taste folgen weitere Einträge.

Programmaufruf

Das Programm verfügt ebenfalls über Hilfsinformationen. Diese rufen Sie durch Eingabe eines Fragezeichens auf. Geben Sie ein

ZDIR ?

erscheint folgender Text:

ZDIR Version 3.00
Copyright (C) 1985, Zenith Data Systems Corporation

Das Dienstprogramm ZDIR zeigt das Inhaltsverzeichnis der Platte mit zusätzlichen Informationen. Es werden alle Dateien angezeigt (auch verdeckte). Vor dem Dateinamen erscheint das Attribut-Byte der Datei. Durch Setzen einzelner Bits wird die Art des Dateieintrages bestimmt. Rufen Sie das Programm ohne Zusatz auf, werden alle Einträge gezeigt.

Eingabe: ZDIR [?] [Dateibez]

dabei bedeutet:

```
?          - es erscheint diese Übersicht
Dateibez   - Name der zu zeigenden Datei und/oder
             Verzeichnis

             - - - - - - - - >> Attribut - Byte << - - - - - - - - -
             --A----- Archiv-Bit    (Datei wurde normal geschrieben)
             ---D---- Directory-Bit (Datei ist Inhaltsverzeichnis)
             ----V--- Volume-Bit    (Datei ist Plattenname)
             -----S-- System-Bit    (Datei ist Systemdatei)
             ------H- Hidden-Bit    (Datei ist verdeckt)
             -------R Read-only-Bit (Datei kann nicht verändert werden)
```

ZDIR

Der Aufruf des Programmes entspricht der Eingabe des internen Befehles DIR. Sie können sich alle Einträge eines Verzeichnisses zeigen lassen oder auch nur bestimmte. Ebenso können Sie den Namen eines Laufwerkes eingeben oder den Namen eines Verzeichnisses. Geben Sie z. B. ein

ZDIR C:\SYSTEM

könnte folgende Übersicht erscheinen:

Laufwerk: C: Name: ZENITH -- 1 Verzeichnis: C:\SYSTEM
14045184 Bytes frei 16699392 Bytes gesamt 24.11.85

Attribut	Name		Größe	Datum	Zeit	Sektor	18:44:20
---D----	.		<DIR>	1.01.80	00:02	225	
---D----	RootDir		<DIR>	1.01.80	00:02	25	
---D----	BAT		<DIR>	24.12.85	12:06	3113	
-D----	LIB		<DIR>	24.12.85	12:06	3121	
----	WS	COM	25600	24.11.85	15:17	361	
----	WSWN	COM	25600	24.12.85	14:24	417	
--A----R	ZDUMP	COM	7665	21.12.85	15:28	473	
--A----R	ZREN	COM	1740	21.12.85	12:50	489	
--A-----	WSMSGS	OVR	27392	1.01.80	03:57	4257	
------H-	ANSI	SYS	1963	15.11.84	16:41	505	
--A----R	ZTYPE	COM	6088	21.12.85	16:20	513	
--A-----	STAT	COM	4091	24.12.85	09:43	2001	
------H-	VDISK	SYS	2565	17.06.85	15:07	537	
--A-----	PART	EXE	24924	31.05.85	15:35	545	
--A-----	ASSIGN	COM	1509	15.05.85	00:00	665	
--A-----	ATTRIB	EXE	7438	15.05.85	00:00	673	
--A-----	CHKDSK	EXE	9296	15.05.85	00:00	689	
--A-----	DEBUG	OM	14677	2.08.85	3:10	145	
-A-----	ELECT	XE	8192	.05.85	2:36	45	
-A-----	ISKCOPY	XE	12192	1.06.85	8:20	61	

▸▸ Betätigen Sie eine Taste für weitere Dateien oder <ESC> für Ende ◂◂

ZDIR

Folgende allgemeine Informationen werden im oberen Teil des Bildes gezeigt:

- Laufwerkname
- Plattenname
- Verzeichnisname
- Gesamtspeicherkapazität des Laufwerkes
- freie Speicherkapazität des Laufwerkes
- Datum und Uhrzeit

Betätigen Sie eine beliebige Taste für weitere Dateien, werden diese Informationen nicht gelöscht. Das Programm löscht den Bildschirm erst ab der vierten Zeile und zeigt dann wieder maximal 18 Einträge.

Das Programm ZDIR zeigt jeden Eintrag des Inhaltsverzeichnisses. Dies gilt auch für verdeckte Dateien und den Plattennamen. In welcher Form MS-DOS Informationen für einen Eintrag speichert, ist auf den Seiten 11.163 und 11.164 erklärt. ZDIR zeigt im Gegensatz zum internen Befehl DIR alle Informationen. Mit Betätigen der Taste ESC kann das Programm vorzeitig verlassen werden.

Fehlermeldungen

Falsche DOS-Version

ERKLÄRUNG:
Sie benutzen das Programm ZDIR nicht unter einer DOS-Version, die 3.10 oder höher ist.

Nicht lizensierte Kopie

ERKLÄRUNG:
Sie benutzen das Programm ZDIR nicht auf einem Zenith PC. Damit ist eine fehlerfreie Funktion nicht möglich.

Falsche Laufwerkbezeichnung

ERKLÄRUNG:
Sie haben das Programm mit einem nicht vorhandenen Laufwerk aufgerufen. Überprüfen Sie Ihre Eingabe.

Datei nicht gefunden

ERKLÄRUNG:
Sie haben das Programm mit einer nicht vorhandenen Datei aufgerufen oder den Dateinamen falsch geschrieben. Überprüfen Sie Ihre Eingabe.

Verzeichnis nicht gefunden

ERKLÄRUNG:
Sie haben das Programm mit einem nicht vorhandenen Verzeichnis aufgerufen oder den Namen falsch geschrieben. Überprüfen Sie Ihre Eingabe.

ZDUMP Extern

Zweck

zeigt den Inhalt einer Binärdatei in hexadezimaler Schreibweise

Syntax

[d:] [Weg] ZDUMP ?
[d:] [Weg] ZDUMP [Dateibez] [/A:]

dabei bedeutet:

d:	vor dem Programmnamen bestimmt das aufzurufende Laufwerk
Weg	vor dem Programmnamen bestimmt das aufzurufende Verzeichnis
?	es erscheint eine Übersicht des Programmes
Dateibez	Name der Datei, deren Inhalt gezeigt werden soll
A:	hexadezimaler Wert für die Startadresse oder das Startsegment

Bemerkungen

Das Programm ZDUMP kann im *Netzwerk* **ohne** Einschränkungen benutzt werden.

Das Programm ZDUMP stellt eine Ergänzung des Programmes DEBUG oder des Monitorprogrammes dar. Es zeigt den Inhalt einer Datei oder eines Speichersegmentes in hexadezimaler Form. Über eine Zusatzeingabe kann für eine Datei eine Startadresse bestimmt werden. Damit müssen Sie eine größere Datei nicht komplett durchblättern, sondern können ab einer beliebigen Stelle beginnen.

Für die Anzeige des Arbeitsspeichers bestimmen Sie mit der Adreßeingabe das Startsegment (Offset ist dann immer Null). Da das Programm nur eine Funktion ausführt, arbeitet es besonders schnell.

Programmaufruf

Das Programm verfügt ebenfalls über Hilfsinformationen. Diese rufen Sie durch Eingabe eines Fragezeichens auf. Geben Sie ein

ZDUMP ?

und es erscheint folgender Text:

ZDUMP Version 3.00
Copyright (C) 1985, Zenith Data Systems Corporation

Das Dienstprogramm ZDUMP zeigt eine Datei oder den Inhalt des Arbeitsspeichers byteweise in hexadezimaler und in dezimaler Schreibweise. Es werden immer 256 (100h) Bytes je Seite gezeigt. Mit den Tasten ‹Pg Dn› und ‹Pg Up› erfolgt ein seitenweises Blättern. Alle Funktionen des Programmes werden über Zusatzeingaben in der Befehlszeile aufgerufen. Ohne Eingabe eines Dateinamens wird der Arbeitsspeicher gezeigt. Dabei entspricht die Startadresse dem zu zeigenden Segment.

Eingabe: ZDUMP [?] [Dateibez] [/A:]

dabei bedeutet:

?	- es erscheint diese Übersicht
Dateibez	- Name der zu zeigenden Datei, mit Laufwerk und/oder Verzeichnis
/A:nnnnn	- Startadresse in hex (maximal 5 Ziffern) oder Datensegment für Speicheranzeige

ZDUMP

Aufruf mit Dateinamen

Rufen Sie das Programm ZDUMP zusammen mit einem Dateinamen auf, wird der Inhalt dieser Datei dargestellt. Dabei darf dem Dateinamen ein Laufwerkname und/oder Verzeichnisname vorangestellt werden. Geben Sie z. B. ein

ZDUMP F:\BIN\ZDUMP.COM

wird die Datei ZDUMP.COM selber dargestellt. Sie ist im Verzeichnis \BIN des Laufwerkes F: gespeichert. Es erscheint folgendes Bild:

DUMP zeigt --> F:\BIN\ZDUMP.COM

Seite 0001

Offset	00	01	02	03	04	05	06	07–08		09	0A	0B	0C	0D	0E	0F	0123456789ABCDEF
00000:	50	E8	D6	05	80	FF	05	74	06	BE	3D	18	E9	E5	05	58	PhV..~.t.>=.ie.X
00010:	E8	29	06	88	26	75	19	E8	05	04	89	1E	93	07	BE	81	h)..&u.h......>.
00020:	00	E8	0B	04	73	03	E9	CB	05	89	3E	95	07	89	36	97	.h..s.iK..>...6.
00030:	07	83	F9	FF	75	05	3D	FF	FF	74	07	03	CF	83	D6	00	..y~u.=~~t..O.V.
00040:	03	C6	89	0E	99	07	A3	9B	07	E8	0E	06	8A	C7	D0	E3	.F....#..h...GPc
00050:	2A	FF	FF	97	75	07	73	03	E9	99	05	E8	FC	05	E9	9C	*~~.u.s.i..h¦.i.
00060:	05	E8	8F	04	73	03	E9	BB	00	A3	85	07	BF	9D	08	E8	.h..s.i;.#..?..h
00070:	BB	01	C7	06	89	07	01	00	8B	1E	85	07	2B	C9	8B	D1	;.G.........+I.Q
00080:	B8	02	42	CD	21	B9	00	01	F7	F1	8B	F0	83	C6	01	BF	8.BM!9..wq.p.F.?
00090:	7D	17	B9	00	01	E8	C4	04	A1	95	07	8B	16	97	07	B9	}.9..hD.!......9
000A0:	00	01	F7	F1	8B	F0	89	36	8D	07	C7	06	89	07	01	00	..wq.p.6..G.....
000B0:	8B	1E	85	07	8B	16	95	07	8B	0E	97	07	B8	00	42	CD	............8.BM
000C0:	21	3D	00	00	75	0B	83	FA	00	75	06	C7	06	89	07	00	!=..u..z.u.G....
000D0:	00	3B	06	99	07	75	0C	3B	16	99	07	75	06	C7	06	89	.;...u.;...u.G..
000E0:	07	02	00	BF	9D	07	B9	80	00	2B	C0	AB	E2	FD	BA	9D	...?..9..+@+b}:.
000F0:	07	B9	00	01	B4	3F	CD	21	3B	C1	73	06	C7	06	89	07	.9..4?M!;As.G...
Offset	00	01	02	03	04	05	06	07–08		09	0A	0B	0C	0D	0E	0F	0123456789ABCDEF

<Esc> = Ende <Pg Dn> = vorwärts Anfang der Datei 0030 Total



Programme der deutschen Version

ZDUMP

Der Bildschirm ist in drei Bereiche aufgeteilt und zeigt folgende Informationen

- Dateiname im oberen Bereich
- Dateiinhalt im mittleren Bereich
- benutzbare Tasten im unteren Bereich

Außer dem kompletten Dateinamen wird im oberen Bereich zusätzlich die aktuelle Seitennummer gezeigt. Dafür wird die Datei in Blöcke von 256 Bytes geteilt, da dies einer Bildschirmseite entspricht. Zusammen mit der Gesamtseitenzahl, die im unteren Bereich des Bildschirmes erscheint, sehen Sie immer die momentane Position in der Datei.

Der mittlere Bereich des Bildschirmes zeigt den Dateiinhalt. Jede Zeile ist in drei Teile gegliedert. Der linke Teil zeigt die Adresse in hexadezimaler Form mit fünf Ziffern. Damit können Dateien bis zu einer Größe von 1 MB dargestellt werden. Dies ist problemlos möglich, da das Programm ZDUMP immer nur 256 Bytes der Datei in den Speicher lädt. Der mittlere Teil der Zeile zeigt den Dateiinhalt in hexadezimaler Schreibweise. Im rechten Teil sehen Sie zusätzlich das entsprechende ASCII-Zeichen. Zur besseren Übersicht werden Graphik- und Steuerzeichen als Punkte dargestellt.

Der untere Bereich des Bildschirmes zeigt die benutzbaren Tasten und die maximale Seitenzahl für diese Datei. Im Programm können drei verschiedene Tasten benutzt werden:

- **PgDn** für vorwärts blättern
- **PgUp** für rückwärts blättern
- **ESC** für Ende

Befinden Sie sich am Anfang oder Ende der Datei, erscheint ein entsprechender Hinweis und die dadurch verdeckte Taste kann nicht mehr benutzt werden.

ZDUMP

Aufruf ohne Dateinamen

Rufen Sie das Programm ZDUMP ohne weiteren Zusatz auf, wird der Inhalt des Arbeitsspeichers dargestellt. Es kann maximal der Inhalt eines Segmentes (von 0000h bis FFFFh) gezeigt werden. Dabei ist der Bildschirm weitgehend identisch mit dem Bildschirm für Dateien. Im oberen Bereich des Bildschirmes erscheint das Startsegment (kann mit dem Zusatz /A bestimmt werden) mit der momentanen Seitennummer. Die maximale Seitenzahl für ein Segment ist immer 256.

/A:n -- Anfangsadresse

Rufen Sie das Programm ZDUMP mit dem Zusatz /A auf, können Sie eine Startadresse in hexadezimaler Form eingeben. Es sind maximal fünf Ziffern möglich. Damit besteht die Möglichkeit, größere Dateien gezielt aufzurufen, ohne daß immer wieder die gesamte Datei durchgeblättert werden muß. Geben Sie z. B. ein:

```
ZDUMP COMMAND:COM /A:4C00
```

Mit dieser Eingabe rufen Sie die Datei COMMAND.COM auf. Die Anzeige beginnt bereits auf Seite 77 mit der Adresse 4C00. Hier sehen Sie die Liste mit den internen Befehlen.

Rufen Sie Dateien mit diesem Zusatz auf, können Sie einen beliebigen hexadezimalen Wert eingeben. Das Programm ZDUMP rundet den Wert jedoch immer auf den nächst niedrigeren Wert, der durch 100H dividiert werden kann.

Rufen Sie das Programm ZDUMP nur mit diesem Zusatz auf, wird der Arbeitsspeicher gezeigt, und der Wert für /A bestimmt das Startsegment. Dann ist nur noch eine vierstellige Zahl erlaubt. Geben Sie z. B. ein:

```
ZDUMP /A:0040
```

Jetzt erscheint der Inhalt des Arbeitsspeichers ab der Adresse 0040:0000. Ab hier werden vom ROM Variable mit Informationen über die Hardware Ihres Zenith PC gespeichert.

Fehlermeldungen

Falsche DOS-Version

ERKLÄRUNG:
Sie benutzen das Programm ZDUMP nicht unter einer DOS-Version, die 3.10 oder höher ist.

Nicht lizensierte Kopie

ERKLÄRUNG:
Sie benutzen das Programm ZDUMP nicht auf einem Zenith PC. Damit ist eine fehlerfreie Funktion nicht möglich.

Falsche Laufwerkbezeichnung

ERKLÄRUNG:
Sie haben das Programm mit einem nicht vorhandenen Laufwerk aufgerufen. Überprüfen Sie Ihre Eingabe.

Datei nicht gefunden

ERKLÄRUNG:
Sie haben das Programm mit einer nicht vorhandenen Datei aufgerufen oder den Dateinamen falsch geschrieben. Überprüfen Sie Ihre Eingabe.

Verzeichnis nicht gefunden

ERKLÄRUNG:
Sie haben das Programm mit einem nicht vorhandenen Verzeichnis aufgerufen oder den Namen falsch geschrieben. Überprüfen Sie Ihre Eingabe.

ZDUMP

Die Datei kann nicht gelesen werden

ERKLÄRUNG:
Das Programm ZDUMP kann die bezeichnete Datei nicht fehlerfrei lesen. Möglicherweise ist die Diskette beschädigt. Überprüfen Sie dies mit CHKDSK.

Falsche Zusatzeingabe

ERKLÄRUNG:
Sie haben das Programm nicht mit dem Zusatz /A aufgerufen. Nur dieser ist erlaubt.

Falscher Wert für /A eingegeben

ERKLÄRUNG:
Sie haben eine Datei mit dem Zusatz /A aufgerufen und einen falschen Hexadezimalwert eingegeben. Überprüfen Sie Ihre Eingabe.

Falscher Wert für Startsegment eingegeben

ERKLÄRUNG:
Sie haben das Programm nur mit dem Zusatz /A aufgerufen. Jetzt ist der maximale Wert FFFFh. Überprüfen Sie Ihre Eingabe.

ZRAM **Extern**

Zweck

zeigt Größe und Aufteilung des Arbeitsspeichers

Syntax

[d:] [Weg] ZRAM [?]

dabei bedeutet:

d:	vor dem Programmnamen bestimmt das aufzurufende Laufwerk
Weg	vor dem Programmnamen bestimmt das aufzurufende Verzeichnis
?	es erscheint eine Übersicht des Programmes

Bemerkungen

Das Programm ZRAM kann im *Netzwerk* **ohne** Einschränkungen benutzt werden.

ZRAM zeigt die Größe des installierten Arbeitsspeichers und seine Aufteilung. Auch dieses Programm kann mit einem Fragezeichen aufgerufen werden und zeigt eine kurze Erklärung. Geben Sie ein

```
ZRAM ?
```

und es erscheint folgender Text:

ZRAM

ZRAM Version 3.00

Das Dienstprogramm ZRAM zeigt jederzeit eine Übersicht des Arbeitsspeichers. Es wird die Größe des gesamten Arbeitsspeichers und die momentane Aufteilung gezeigt. Dabei wird zwischen dem Systemspeicher und eventuellen Erweiterungen unterschieden. Erweiterungsprogramme werden immer nach dem Betriebssystem geladen. Dies ist zuerst der Befehlsinterpreter (die Datei COMMAND.COM). Danach können z. B. Programme zur Tastaturanpassung oder für den Hintergrunddruck permanent geladen werden.

Der freie Arbeitsspeicher kann von aufzurufenden Anwendungsprogrammen benutzt werden. Das Programm ZRAM wird durch Eingabe seines Namens aufgerufen.

Rufen Sie das Programm ohne weiteren Zusatz auf, zeigt es den Arbeitsspeicher in folgender Form:

```
655360 Bytes Gesamtarbeitsspeicher
 47952 Bytes Systemspeicher
  8832 Bytes Systemerweiterungen

598576 Bytes freier Arbeitsspeicher
```

In der obersten Zeile sehen Sie die maximale Speichergröße, die MS-DOS momentan bearbeiten kann. In der zweiten Zeile wird die Größe des Systemspeichers gezeigt. Dieser Bereich kann von keinem Anwenderprogramm benutzt oder verändert werden. Hier sind die Systemdateien IO.SYS und MS-DOS.SYS gespeichert. Zusätzliche Modifikationen der Datei CONFIG.SYS (Zwischenspeicher oder Gerätetreiber) sind hier ebenfalls gespeichert. Die dritte Zeile zeigt die Größe der installierten Systemerweiterungen. Hierzu zählen die Datei COMMAND.COM (nur residenter Teil) und permanent geladene Programme. Dies kann z. B. KEYBGR oder PRINT sein.

Der noch verbleibende freie Speicher kann von jedem beliebigen Programm genutzt werden.

ZREN Extern

Zweck

ändert den Namen eines Eintrags im Inhaltsverzeichnis

Syntax

[d:] [Weg] ZREN ?
[d:] [Weg] ZREN Dateibez Dateiname

dabei bedeutet:

d:	vor dem Programmnamen bestimmt das aufzurufende Laufwerk
Weg	vor dem Programmnamen bestimmt das aufzurufende Verzeichnis
?	es erscheint eine Übersicht des Programmes
Dateibez	Name des Eintrags, der geändert werden soll
Dateiname	neuer Name für den bezeichneten Eintrag

Bemerkungen

Das Programm ZREN kann im *Netzwerk* **nicht** benutzt werden.

Das Programm erweitert den internen MS-DOS-Befehl RENAME oder REN. Es kann nicht nur einen Dateinamen sondern generell jeden Eintrag eines Inhaltsverzeichnisses ändern. Dies kann sein:

- Dateiname
- Verzeichnisname
- Plattenname

ZREN

Mit diesem Programm *ändern* Sie nur den Namen eines Eintrags. Der Eintrag selber wird nicht kopiert oder auf der Diskette an einen anderen Platz geschrieben. Da mit diesem Programm jeder Eintrag geändert werden kann, sollte es vorsichtig benutzt werden.

ACHTUNG:
Sie können dieses Programm nicht über das Netzwerk benutzen. Ebenso können Sie damit nicht Einträge auf einem umgeleiteten (mit ASSIGN bestimmt) oder auf einem vorübergehenden Laufwerk (mit SUBST bestimmt) umbenennen.

Programmaufruf

Das Programm verfügt ebenfalls über Hilfsinformationen. Diese rufen Sie durch Eingabe eines Fragezeichens auf. Geben Sie ein

ZREN ?

und es erscheint folgender Text:

ZREN Version 3.00
Copyright (C) 1985, Zenith Data Systems Corporation

Das Dienstprogramm ZREN ergänzt den internen Befehl REN oder RENAME. Es kann jeden Eintrag eines Inhaltsverzeichnisses umbenennen, also auch Platten- und Verzeichnisnamen. Das Programm wird mit der gleichen Eingabe aufgerufen wie der interne Befehl. Benutzen Sie für den Originalnamen Globalzeichen (* oder ?), können nur Dateinamen oder nur Verzeichnisnamen geändert werden. Das Programm ZREN kann nicht im Netzwerk oder für ein umgeleitetes oder ersetztes Laufwerk benutzt werden.

Eingabe: ZREN [?] Dateibez Dateiname

dabei bedeutet:

? - es erscheint diese Übersicht
Dateibez - alter Name des zu bearbeitenden Eintrags
Dateiname - neuer Name des zu bearbeitenden Eintrags

Der erste Dateiname nach dem Befehlsnamen ist immer der Name des umzubenennenden Eintrages. Diesem Namen kann eine Laufwerkbezeichnung vorangestellt werden. Der zweite Name ist der neue Name des Eintrages. Diesem Namen braucht keine Laufwerkbezeichnung und auch nicht der Name eines Verzeichnisses vorangestellt werden. Erfolgt dennoch eine Eingabe, kann das Programm nicht arbeiten. Achten Sie darauf, daß der neue Dateiname noch nicht auf der Diskette vergeben ist. In diesem Falle erscheint folgende Fehlermeldung:

```
Verzeichnis/Dateiname doppelt oder nicht vorhanden
```

Sie können mit dem Befehl mehrere Einträge zugleich umbenennen, wenn Sie im Originalnamen Globalzeichen verwenden. Das Programm kann aber nicht Datei- und Verzeichnisnamen gleichzeitig umbenennen.

Einen Plattennamen können Sie umbenennen, wenn Sie im Root-Dir der Platte sind oder dem Namen einen umgekehrten Schrägstrich voranstellen. Dies ist aber auch mit dem Programm LABEL möglich.

Beispiele

Ist im Rootdir der Platte das Verzeichnis BUCH gespeichert und soll umbenannt werden, geben Sie ein:

```
ZREN BUCH ALT
```

Das Verzeichnis hat jetzt den Namen ALT. Geben Sie den Befehl DIR oder ZDIR ein, sehen Sie die Änderung.

ZREN

Verwenden Sie den Befehl ZREN mit Globalzeichen (Stern oder Fragezeichen), werden sämtliche erfaßten Dateinamen gleichzeitig umbenannt. Die Globalzeichen müssen bei beiden Namen an gleicher Stelle im Dateinamen stehen. Es werden dann diese Zeichen nicht verändert. Geben Sie ein:

```
ZREN *.LST *.DOC
```

Sämtliche Dateien mit dem Namenszusatz .LST erhalten jetzt den Namenszusatz .DOC. Der Dateiname wird dabei nicht verändert. Auch hier wäre es möglich, vor dem ersten Namen eine Laufwerkbezeichnung und/oder einen Verzeichnisnamen einzugeben.

Fehlermeldungen

Falsche DOS-Version

ERKLÄRUNG:
Sie benutzen das Programm ZREN nicht unter einer DOS-Version, die 3.10 oder höher ist.

Nicht lizensierte Kopie

ERKLÄRUNG:
Sie benutzen das Programm ZREN nicht auf einem Zenith PC. Damit ist eine fehlerfreie Funktion nicht möglich.

Falsche Laufwerkbezeichnung

ERKLÄRUNG:
Sie haben das Programm mit einem nicht vorhandenen Laufwerk aufgerufen. Überprüfen Sie Ihre Eingabe.

Datei nicht gefunden

ERKLÄRUNG:
Sie haben das Programm mit einer nicht vorhandenen Datei aufgerufen oder den Dateinamen falsch geschrieben. Überprüfen Sie Ihre Eingabe.

Verzeichnis nicht gefunden

ERKLÄRUNG:
Sie haben das Programm mit einem nicht vorhandenen Verzeichnis aufgerufen oder den Namen falsch geschrieben. Überprüfen Sie Ihre Eingabe.

Ein umgeleitetes oder ersetztes Laufwerk kann nicht bearbeitet werden

ERKLRÄRUNG:
Sie haben das Programm ZREN mit einem Laufwerknamen aufgerufen, der mit dem Programm ASSIGN umgeleitet wurde oder mit dem Programm SUBST ein Verzeichnis ersetzt. Dies ist nicht erlaubt.

Ein Laufwerk des Netzwerkes kann nicht bearbeitet werden

ERKLÄRUNG:
Sie haben versucht, mit dem Programm ZREN einen Eintrag auf einem Laufwerk des Netzwerkes zu ändern. Dies ist nicht möglich.

ZTYPE Extern

Zweck

zeigt den Inhalt einer ASCII-Datei seitenweise an

Syntax

[d:] [Weg] ZTYPE ?
[d:] [Weg] ZTYPE [/X] < Dateibez [> Dateibez]

dabei bedeutet:

d: vor dem Programmnamen bestimmt das aufzurufende Laufwerk

Weg vor dem Programmnamen bestimmt das aufzurufende Verzeichnis

? es erscheint eine Übersicht des Programmes

< **Dateibez** Name der Datei, deren Inhalt gezeigt werden soll

> **Dateibez** Name einer Ausgabedatei oder eines Ausgabegerätes

/X Kombination aus folgenden Zusatzeingaben:

- **/D** Ausgabe in eine Datei oder ein Peripheriegerät
- **/W** zeigt formatierte WordStar-Dateien als direkt lesbaren Text

Bemerkungen

Das Programm ZTYPE kann im *Netzwerk* **ohne** Einschränkungen benutzt werden.

Das Programm ZTYPE ist ein Filter, der ASCII-Daten der Standardeingabe liest, bearbeitet und zum Standardausgabegerät sendet. Damit gleicht er dem Filter MORE. Das Programm verfügt aber über Zusatzfunktionen.

Programmaufruf

Das Programm verfügt ebenfalls über Hilfsinformationen. Diese rufen Sie durch Eingabe eines Fragezeichens auf. Geben Sie ein:

ZTYPE ?

und es erscheint folgender Text:

ZTYPE Version 3.00
Copyright (C) 1985, Zenith Data Systems Corporation

Das Dienstprogramm ZTYPE arbeitet als Filter und gibt Daten seitenweise auf den Bildschirm aus. Der Filter liest die Standardeingabe. Nach Ausgabe von 24 Zeilen stoppt das Programm, und erst nach Betätigen einer beliebigen Taste wird die Ausgabe fortgesetzt. Das Programm ZTYPE eignet sich besonders zur Verkettung von Befehlen. Die Zusatzeingabe /W (ordstar) formt die Zeichen der Dateien für die Ausgabe auf den Bildschirm in lesbaren ASCII-Kode um. Die Zusatzeingabe /D gibt den Text in eine Datei aus. Dies kann eine beliebige Diskettendatei oder ein Peripheriegerät sein.

Eingabe: ZTYPE [?] [/W] [/D] < Dateibez [> Dateiname]

dabei bedeutet:

?	- es erscheint diese Übersicht
Dateibez	- die Daten dieser ASCII-Datei werden ausgegeben
Dateiname	- Name der Datei für die umgeleiteten Daten
/D	- der Text wird in eine Datei umgeleitet
/W	- WordStar-Dateien werden in lesbare Texte umgewandelt
< >	- Zeichen zur Datenumleitung

ZTYPE

Mit diesem Filter können Sie größere Datenmengen seitenweise auf den Bildschirm ausgeben. Dabei löscht der Filter vor jeder neuen Ausgabe zuerst den Bildschirm und beschreibt ihn dann von oben beginnend. Dies geschieht sehr schnell, da das Betriebssystem dafür nicht benutzt wird. Am Ende einer jeden Seite erscheint die Zeile

```
ZTYPE: Betätigen Sie eine Taste für mehr Text, <ESC> für Ende
```

und die Ausgabe wird unterbrochen. Erst nach Betätigen einer beliebigen Taste wird die Ausgabe fortgesetzt. Wie jeder andere Filter auch, liest ZTYPE erst Daten ein, bevor es die Daten anzeigt. Die Eingabe erfolgt über das Standardeingabegerät. Ein Beispiel:

```
ZTYPE BRIEF.TXT
```

Der Inhalt der Datei BRIEF.TXT erscheint auf dem Bildschirm. Ist es eine größere Datei, wird die Ausgabe nach jeder vollen Seite angehalten und erst nach Betätigen einer beliebigen Taste fortgesetzt.

Sie können jetzt, falls gewünscht, einzelne Seiten der Datei mit der Tastenkombination **SHIFT-PRTSC** auf den Drucker ausgeben. Nähere Einzelheiten dazu im Kapitel 8.

/D -- Ausgabe in Datei

Soll die Ausgabe der Daten nicht auf den Bildschirm erfolgen, kann mit dem Zeichen für Datenumleitung (>) ein anderes Ausgabegerät bestimmt werden. Damit der Filter ZTYPE dies auch erkennt, muß er mit dem Zusatz /D aufgerufen werden. Nur dann werden die Daten mit Hilfe des Betriebssystemes umgeleitet. Auf dem Bildschirm erscheint ein entsprechender Hinweis.

Ein Beispiel:

ZTYPE /D < BRIEF.TXT > PRN

Der Filter liest die Datei BRIEF.TXT und gibt sie auf den Systemdrucker aus. Die Datei wird komplett gelesen und geschrieben, es erfolgt keine seitenweise Ausgabe. Auf dem Bildschirm erscheint die Meldung:

Die Daten werden in eine Datei geschrieben

Der Zusatz /D kann mit dem Zusatz /W gleichzeitig benutzt werden.

/W -- Ausgabe von WordStar-Dateien

Das Programm WordStar benutzt zur Formatierung des Textes ein besonderes Format. Damit werden keine reinen ASCII-Dateien erstellt, die direkt lesbar sind. Suchen Sie z. B. einen bestimmten Text und müssen mehrere Dateien lesen, dauert es mit dem Programm einige Zeit. Benutzen Sie dann den Filter ZTYPE mit dem Zusatz /W. Der Filter liest die Datei ebenfalls und konvertiert den Inhalt dabei. So kann anschließend der Text gelesen werden. Ein Beispiel:

ZTYPE /W < BRIEF.DOC

Die Datei BRIEF.DOC wird gelesen und auf dem Bildschirm dargestellt. Benutzen Sie gleichzeitig den Zusatz /D, können Sie den Text konvertieren und im reinen ASCII-Format speichern oder direkt ausdrucken. Ein Beispiel:

ZTYPE /D/W < BRIEF.DOC > LPT1

Der Filter ZTYPE liest die Datei BRIEF.DOC und sendet sie zum Ausgang LPT1.

ZTYPE

Fehlermeldungen

Falsche DOS-Version

ERKLÄRUNG:
Sie benutzen das Programm ZTYPE nicht unter einer DOS-Version, die 3.10 oder höher ist.

Nicht lizensierte Kopie

ERKLÄRUNG:
Sie benutzen das Programm ZTYPE nicht auf einem Zenith PC. Damit ist eine fehlerfreie Funktion nicht möglich.

ZSTATS Extern

Zweck

zeigt allgemeine Informationen eines Laufwerkes

Syntax

[d:] [Weg] ZSTATS ?
[d:] [Weg] ZSTATS [d:]

dabei bedeutet:

d:	vor dem Programmnamen bestimmt das aufzurufende Laufwerk
Weg	vor dem Programmnamen bestimmt das aufzurufende Verzeichnis
?	es erscheint eine Übersicht des Programmes
d:	nach dem Programmnamen bezeichnet das zu bearbeitende Laufwerk

Bemerkungen

Das Programm ZSTATS kann im *Netzwerk* **ohne** Einschränkungen benutzt werden.

Mit diesem Dienstprogramm können Sie statistische Daten eines Laufwerkes erfragen. Dies ist z. B. wichtig bei unbekannten Disketten oder nach einer Neuaufteilung des Winchesterlaufwerkes.

ZSTATS

Das Programm verfügt ebenfalls über Hilfsinformationen. Diese rufen Sie durch Eingabe eines Fragezeichens auf. Geben Sie ein

ZSTATS ?

und es erscheint folgender Text:

```
ZSTATS Version 3.00
Copyright (C) 1985, Zenith Data Systems Corporation

Das Dienstprogramm ZSTATS zeigt allgemeine Informationen eines bestimmten
Laufwerkes. Dies ist z. B. erwünscht bei einer unbekannten Diskette oder nach einer
Aufteilung eines Winchesterlaufwerkes. Außerdem kann es zur Diagnose bei nicht
auffindbaren Plattenfehlern benutzt werden.

Das Programm kann im Netzwerk benutzt werden, jedoch nicht für umgeleitete
Laufwerke (ASSIGN) oder Laufwerke, die ein Verzeichnis ersetzen (SUBST).

Eingabe: ZSTATS [?] [d:]

dabei bedeutet:

?          - es erscheint diese Übersicht
d:         - Name des zu bearbeitenden Laufwerkes
```

ZSTATS

Rufen Sie mit dem Programm Informationen über ein bestimmtes Laufwerk ab, geben Sie z. B. ein:

ZSTATS C:

und es könnte folgendes Bild erscheinen:

```
                         Platteninformation

Laufwerk: C:   Name: ZENITH -- 1   Verzeichnis: C:\BUCH

Formatiert von : ZDS 3.0

Speicher gesamt: 16699392 Bytes     F.A.T. ID-Byte.............:  F8
Speicher frei......: 13631488 Bytes     F.A.T. Anzahl................:   2 Stück
                                        F.A.T. Größe.................:  12 Sektoren
Sektorgröße........:      512 Bytes     F.A.T. #1 beginnt........:   1 Sektor
Sektorzahl..........:    32616 Sektoren
Sektor/Spur........:       17 Sektoren  ROOTDIR Einträge.....: 512 Stück
Sektor/Cluster...:        8 Sektoren    ROOTDIR beginnt.......:  25 Sektor

Cluster gesamt..:     4077 Cluster      Daten beginnen............:  57 Sektor
Clustergröße......:     4096 Bytes
Cluster frei........:     3328 Cluster  Zahl der Köpfe..............:   3 Köpfe

                     ››› <ESC> für Ende ‹‹‹
```

ZSTATS

Es werden insgesamt 21 verschiedene Informationen über das Laufwerk gezeigt. Dies sind im einzelnen:

- **Laufwerkname**
  Name des bearbeiteten Laufwerkes (A: bis maximal Z:)
- **Plattenname**
  wird nur gezeigt sofern vorhanden
- **Verzeichnisname**
  Name des aktuellen Verzeichnisses
- **Formatiert von:**
  Dieser Name wird vom Programm FORMAT auf das Laufwerk geschrieben
- **Gesamtspeicherkapazität des Laufwerkes**
  maximale Speicherkapazität für Daten in Bytes
- **freie Speicherkapazität des Laufwerkes**
  momentan noch freie Kapazität für neue Daten in Bytes
- **Sektorgröße**
  unter MS-DOS immer 512 Bytes
- **Sektorzahl**
  Zahl der insgesamt vorhandenen Sektoren (unabhängig von ihrer Verwendung)
- **Sektor/Spur**
  Zahl der Sektoren pro Spur, die Tabelle auf Seite 11.153 gibt eine Übersicht
- **Sektor/Cluster**
  Zahl der Sektoren pro Cluster. MS-DOS speichert Diskettendaten nicht in Sektoren sondern in Sektorgruppen (Clustern).
- **Cluster gesamt**
  Zahl der maximalen Cluster pro Laufwerk. Dies wäre die maximale Zahl der Dateien, die auf dem Laufwerk gespeichert werden können.

- **Clustergröße**
  Größe einer Sektorengruppe in Bytes. Dieser Platz wird mindestens pro Datei belegt. Im Durchschnitt bleibt die Hälfte der Bytes pro Datei ungenutzt.

- **Cluster frei**
  Zahl der noch freien Sektorengruppen für neue Daten

- **FAT ID-Byte**
  Mit diesem Byte erkennt MS-DOS den Plattentyp. Die Tabelle auf Seite 11.159 zeigt eine Übersicht.

- **Zahl der vorhandenen FAT**
  MS-DOS speichert aus Sicherheitsgründen immer zwei

- **Größe eines FAT in Sektoren**
  Die Größe variiert je nach Speicherkapazität des Laufwerkes. Je größer das FAT jedoch ist, desto länger dauert es, bis eine Datei gelesen werden kann.

- **erster Sektor des ersten FAT**
  Dies ist meistens der Sektor zwei (Im ersten Sektor ist der BOOT-LOADER gespeichert)

- **maximale Zahl der Einträge im Rootdir**
  Dieser Wert variiert je nach Plattengröße von 64 Einträgen (einseitig formatierte Disketten) bis 512 Einträgen (Winchesterlaufwerk)

- **erster Sektor des Rootdir**
  Dieser Wert ändert sich mit der Größe der FAT

- **erster Datensektor des Laufwerkes**
  Ab diesem Sektor werden Daten gespeichert. Die Datei IO.SYS muß immer auf diesem Sektor beginnen, damit es bootfähig ist (kann mit dem Programm ZDIR überprüft werden).

- **Zahl der Schreib-Lese-Köpfe des Laufwerkes**
  Diskettenlaufwerke verfügen über ein oder zwei Köpfe, Winchesterlaufwerk über drei oder mehr

ZSTATS

Fehlermeldungen

Falsche DOS-Version

ERKLÄRUNG:
Sie benutzen das Programm ZSTATS nicht unter einer DOS-Version, die 3.10 oder höher ist.

Nicht lizensierte Kopie

ERKLÄRUNG:
Sie benutzen das Programm ZSTATS nicht auf einem Zenith PC. Damit ist eine fehlerfreie Funktion nicht möglich.

Falsche Laufwerkbezeichnung

ERKLÄRUNG:
Sie haben das Programm mit einem nicht vorhandenen Laufwerk aufgerufen. Überprüfen Sie Ihre Eingabe.

Ein umgeleitetes oder ersetztes Laufwerk kann nicht bearbeitet werden

ERKLRÄRUNG:
Sie haben das Programm ZSTATS mit einem Laufwerknamen aufgerufen, der mit dem Programm ASSIGN umgeleitet wurde oder mit dem Programm SUBST ein Verzeichnis ersetzt. Dies ist nicht erlaubt.

Index

Index

Index

Index

Index

Index

Index

Index

Index

E

Index

Index

Index

Index

Index